HPI Future SOC Lab: Proceedings 2017

Christoph Meinel, Andreas Polze, Karsten Beins, Rolf Strotmann, Ulrich Seibold, Kurt Rödszus, Jürgen Müller (Eds.)

Technische Berichte Nr. 130

des Hasso-Plattner-Instituts für
Digital Engineering an der Universität Potsdam

Universität Potsdam

HPI Hasso Plattner Institut
Digital Engineering · Universität Potsdam

Technische Berichte des Hasso-Plattner-Instituts für
Digital Engineering an der Universität Potsdam

Technische Berichte des Hasso-Plattner-Instituts für
Digital Engineering an der Universität Potsdam | 130

Christoph Meinel | Andreas Polze | Karsten Beins | Rolf Strotmann |
Ulrich Seibold | Kurt Rödszus | Jürgen Müller (Eds.)

HPI Future SOC Lab

Proceedings 2017

Universitätsverlag Potsdam

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind
im Internet über http://dnb.dnb.de/ abrufbar.

Universitätsverlag Potsdam 2019
http://verlag.ub.uni-potsdam.de/

Am Neuen Palais 10, 14469 Potsdam
Tel.: +49 (0)331 977 2533 / Fax: 2292
E-Mail: verlag@uni-potsdam.de

Die Schriftenreihe **Technische Berichte des Hasso-Plattner-Instituts für Digital Engineering an der Universität Potsdam** wird herausgegeben von den Professoren des Hasso-Plattner-Instituts für Digital Engineering an der Universität Potsdam.

ISSN (print) 1613-5652
ISSN (online) 2191-1665

Online veröffentlicht auf dem Publikationsserver der Universität Potsdam
https://doi.org/10.25932/publishup-43310
https://nbn-resolving.org/urn:nbn:de:kobv:517-opus4-433100

Zugleich gedruckt erschienen im Universitätsverlag Potsdam:
ISBN 978-3-86956-475-3

Preface

The *HPI Future SOC Lab* is a cooperation of the Hasso Plattner Institute (HPI) and industry partners. Its mission is to enable and promote exchange and interaction between the research community and the industry partners.

The HPI Future SOC Lab provides researchers with free of charge access to a complete infrastructure of state of the art hard and software. This infrastructure includes components, which might be too expensive for an ordinary research environment, such as servers with up to 64 cores and 2 TB main memory. The offerings address researchers particularly from but not limited to the areas of computer science and business information systems. Main areas of research include cloud computing, parallelization, and In-Memory technologies.

This technical report presents results of research projects executed in 2017. Selected projects have presented their results on April 25th and November 15th 2017 at the Future SOC Lab Day events.

Contents

Prof. Dr. Lars Lundberg, Blekinge Institute of Technology

Dr. Igor Di Varano, Leibniz Institut für Astrophysik, Potsdam

Prof. Dr. Jan Eloff, University of Pretoria, South Africa

PD Dr. Benjamin Fabian, Humboldt University of Berlin

Prof. Wei Lu, Singapore University of Technology and Design

Dr. Luis Ángel Trejo Rodríguez, Instituto Tecnológico y de Estudios Superiores de Monterrey, Mexico

Dr. Ernst Niederleithinger, Federal Institute of Materials Research and Testing (BAM), Germany

Fall 2017

Prof. Dr. Andreas Polze, Hasso Plattner Institute

Prof. Dr. Lars Lundberg, Blekinge Institute of Technology

Prof. Dr. Tobias Friedrich, Hasso Plattner Institute

Dr. Luis Ángel Trejo Rodríguez, Instituto Tecnológico y de Estudios Superiores de Monterrey, Mexico

Prof. Dr. Jan Eloff, University of Pretoria, South Africa

Prof. Wei Lu, Singapore University of Technology and Design

Dr. Matthias Uflacker, Hasso Plattner Institute

Prof. Dr. Christoph Meinel, Hasso Plattner Institute

Prof. Dr. Roberto C. L. de Oliveira, Federal University of Para

Dr.-Ing. André van Hoorn, University of Stuttgart, Institute of Software Technology

Prof. Dr. Helmut Krcmar, Technical University of Munich

Prof. Dr. Bernhard Kuster, Technical University of Munich

Efficient Stream Processing on Multi-Core Processors: Relative Location aware Scheduling

Shuhao Zhang

National University of Singapore
shuhao.zhang@comp.nus.edu.sg

1 Project Idea

Data stream processing (DSP) systems such as Apache Flink [4], Apache Storm [6], and Apache Samza [5] have recently gained much attention owing to their ability to process huge volumes of data with low latency. Streaming applications (i.e., jobs) on DSP systems are commonly represented by directed acyclic graphs (DAG) where vertices represent operators, and edges represent the data dependencies between operators. A fundamental problem in modern DSP systems is how to allocate (i.e., schedule) operators of a job into the physical resources (e.g., compute node) in order to achieve certain optimization goals, such as maximize throughput, minimize latency or minimize resource consumption, etc. Many research efforts are devoted to this problem (e.g., [3, 9, 14, 24]).

On the other hand, modern machines scale to multiple sockets, and non-uniform memory access (NUMA) becomes an important performance factor for data management systems (e.g., [19], [20]). For example, recent NUMA systems have already supported hundreds of CPU cores and multi-terabytes of memory [26]. However, state-of-the-art DSP systems are mainly designed and optimized for scaling out using a cluster of commodity machines (e.g., [3, 22, 28]), and existing optimization techniques overlook the effect of NUMA of a single sever.

In this paper, we identify that it is a non-trivial task for scheduling stream processing on NUMA systems.

First, due to the design of pipelined processing with message passing on modern DSPs, message passing cost (and the corresponding resource consumption) on each operator is significant, and depends on the *relative* location (i.e., on the same CPU socket, or on different CPU sockets) between it and its producers [29]. As a result, changing the location of one operator affects the processing rate (determined by both message passing cost and execution cost) of its consumers, which may require to be rescheduled.

Second, in order to compare different scheduling plans, a common requirement is a good cost estimation technique. However, existing approaches often assume the message passing cost (and the corresponding resource demand) is fixed and independent of scheduling results (e.g., [3, 9, 10, 14, 15]). Although this assumption may fit well and continuously to be valid in the context of distributed stream processing or traditional store-and-process query processing (e.g., [15] assumed resource demand of each operator to be independent of scheduling results), it may not apply for stream processing on multi-core processors under the non-uniform memory access (NUMA) effect.

To allow efficient stream processing on multi-core processors, a new scheduling approach is needed. In this paper, we propose a novel *relative location aware scheduling* (RLS) paradigm for stream processing on multi-core processors. Specifically, RLS generates both operator allocation and stream partition plans in a NUMA-aware manner that maximize application output rate. Different from existing approaches, RLS is a more fine-grained approach that considers relative location of each pair of producer and consumer while searching for optimal scheduling plan. Furthermore, it generates both operator placement plan and stream partition plans that work in combination for better global execution efficiency.

We adopt and extend the rate-based model [21, 27] to guide the optimization. Specifically, we use a) input, b) process and c) output rate of each operator to denote the average number of tuples it able to a) receive, b) process, and c) generate in a unit of time. We make necessary extensions on the model in order to accurately characterize the behavior of an operator in different scheduling plans under NUMA effect. In particular, we propose to break down the cost in handling each input data of an operator into the following components: 1) the cost to fetch the input data, 2) the cost spent in in-core computation, and 3) the cost spent in out-of-core computation (i.e., mainly due to memory access on private data structure). Those components have different impacts on the physical infrastructure (i.e., CPU, memory bandwidth, and QPI bandwidth) and may (i.e., data access cost) or may not (i.e., the other two) vary in different scheduling plans. In this way, we can accurately predict the input, process and output rate of each operator and compare different scheduling plans under the NUMA effect.

2 Experiment Settings

2.1 Resource Used

We are currently using an eight-sockets server with the Intel Xeon Nehalem EX X7560 processors from HPI. Table 1 shows the detailed specification of our testing environment.

2.2 Micro Benchmark

We design our streaming benchmark according to the four criteria proposed by Jim Gray [16]. As a start, we design the benchmark consisting of seven streaming applications including Stateful Word Count (WC), Fraud Detection (FD), Spike Detection (SD), Traffic Monitoring (TM), Log Processing (LG), Spam Detection in VoIP (VS), and Linear Road (LR).

We briefly describe how they achieve the four criteria. 1) Relevance: the applications cover a wide range of memory and computational behaviors, as well as different

Table 1: Detailed specification on our testing environment

Component		Description
Processor		Intel Xeon X7560, Nehalem EX
	Cores (per socket)	8 * 2.27GHz (hyper-threading disabled)
	Sockets	8
	L1 cache	32KB Instruction, 32KB Data per core
	L2-Cache	256KB per core
	Last level cache	24567KB per socket
Memory		8 * 256GB, Quard DDR3 channels, 800 MHz
Java HotSpot VM		java 1.8.0_77, 64-Bit Server VM, (mixed mode) -server -XX:+UseG1GC -XX:+UseNUMA

application complexities so that they can capture the DSP systems on scale-up architectures; 2) Portability: we describe the high-level functionality of each application, and they can be easily applied to other DSP systems; 3) Scalability: the benchmark includes different data sizes; 4) Simplicity: we choose the applications with simplicity in mind so that the benchmark is understandable.

Our benchmark covers different aspects of application features. *First*, our applications cover different runtime characteristics. Specifically, TM has highest CPU resource demand, followed by LR, VS and LG. CPU resource demand of FD and SD is relatively low. The applications also have variety of memory bandwidth demands. *Second*, topologies of the applications have various structural complexities. Specifically, WC, FD, SD, and TM have single chain topologies, while LG, VS, and LR have complex topologies. Figure 2 shows the topologies of the seven applications.

In the following, we describe each application including its application scenario, implementation details and input setup. In all applications, we use a simple sink operator to measure the throughput.

Stateful Word Count (WC): The stateful word-count counts and remembers the frequency of each received word unless the application is killed. The topology of WC is a single chain composed of a Split operator and a Count operator. The Split operator parses sentences into words and the Count operator reports the number of occurrences for each word by maintaining a hashmap. This hashmap is once created in the initialization phase and is updated for each receiving word. The input data of WC is a stream of string texts generated according to a Zipf-Mandelbrot distribution (skew set to 0) with a vocabulary based on the dictionary of Linux kernel (3.13.0-32-generic).

Fraud Detection (FD): Fraud detection is a particular use case for a type of problems known as outliers detection. Given a transaction sequence of a customer, there is a probability associated with each path of state transition, which indicates the chances of fraudulent activities. We use a detection algorithm called *missProbability* [13] with sequence window size of 2 events. The topology of FD has only one operator, named as Predict, which is used to maintain and update the state transition of each customer. We use a sample transaction with 18.5 million records for testing. Each record includes customer ID, transaction ID, and transaction type.

Log Processing (LG): Log processing represents the streaming application of performing real-time analyzing on system logs. The topology of LG consists of four operators. The Geo-Finder operator finds out the country and city where an IP request is from, and the Geo-Status operator maintains all the countries and cities that have been found so far. The Status-Counter operator performs statistics calculations on the status codes of HTTP logs. The Volume-Counter operator counts the number of log events per minute. We use a subset of the web request data (with 4 million events) from the 1998 World Cup Web site [11]. For data privacy protection, each actual IP address in the requests is mapped to randomly generated but fixed IP address.

Spike Detection (SD): Spike detection tracks measurements from a set of sensor devices and performs moving aggregation calculations. The topology of SD has two operators. The Moving-Average operator calculates the average of input data within a moving distance. The Spike-Detection operator checks the average values and triggers an alert whenever the value has exceeded a threshold. We use the Intel lab data (with 2 million tuples) [17] for this application. The detection threshold of moving average values is set to 0.03.

Spam Detection in VoIP (VS): Similar to fraud detection, spam detection is a use case of outlier detection. The topology of VS is composed of a set of filters and modules that are used to detect telemarketing spam in Call Detail Records (CDRs). It operates on the fly on incoming call events (CDRs), and keeps track of the past activity implicitly through a number of on-demand time-decaying bloom filters. A detailed description of its implementation can be found at [8]. We use a synthetic data set with 10 million records for this application. Each record contains data on a calling number, called number, calling date, answer time, call duration, and call established.

Traffic Monitoring (TM): Traffic monitoring performs real-time road traffic condition analysis, with real-time mass GPS data collected from taxis and buses. TM contains a Map-Match operator which receives traces of an object (e.g., GPS loggers and GPS-phones) including altitude, latitude, and longitude, to determine the location (regarding a road ID) of this object in real-time. The Speed-Calculate operator uses the road ID result generated by Map-Match to update the average speed record of the corresponding road. We use a subset (with 75K events) of GeoLife GPS Trajectories [12] for this application.

Linear Road (LR): Linear Road (LR) is used for measuring how well a DSP system can meet real-time query response requirements in processing a large volume of streaming and historical data [7]. It models a road toll network, in which tolls depend on the time of the day and level of congestions. Linear Road has been used by many DSP systems, e.g., Aurora [2], Borealis [1], and System S [18]. LR produces reports of the account balance, assessed tolls on a given expressway on a given day, or estimates cost for a journey on an expressway. We have followed the implementation of the previous study [25] for LR. Several queries specified in LR are implemented as operators and integrated into a single topology. The input to LR is a continuous stream of position reports and historical query requests. We merge the two data sets obtained from [23] resulting in 30.2 million input records (including both position reports and query requests) and 28.3 million historical records.

3 Findings

In this section, we first present performance evaluation results of different applications on Storm and Flink on multi-core processors from one of our previous studies [29] as the motivation. Then, we show new findings of current study based on our newly designed system and optimization techniques.

3.1 Motivation

We tune each application on both Storm and Flink according to their specifications such as the number of threads in each operator.

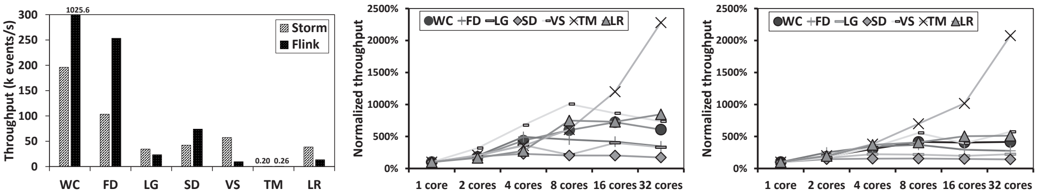
Throughput on a single socket. Figure 1a shows the throughput of running different applications on Storm and Flink on a single CPU socket. The comparison between Storm and Flink is inconclusive. Flink has higher throughput than Storm on WC, FD, and SD, while Storm outperforms Flink on VS and LR. The two systems have similar throughput on TM and LG.

Scalability on varying number of CPU cores. We vary the number of CPU cores from 1 to 8 on the same CPU socket and then vary the number of sockets from 2 to 4 (the number of CPU cores from 16 to 32). Figures 1b and 1c show the normalized throughput of running different applications with varying number of cores/sockets on Storm and Flink, respectively. The performance results are normalized to their throughputs on a single core.

We have the following observations. *First*, on a single socket, most of the applications scale well with the increasing number of CPU cores for both Storm and Flink. *Second*, most applications perform only slightly better or even worse on multiple sockets than on a single socket. FD and SD become even worse on multiple sockets than on a single socket, due to their relatively low compute resource demand. Enabling multiple sockets only brings additional overhead of remote memory accesses. WC, LG and VS perform similarly for different numbers of sockets. The throughput of LR increases marginally with the increasing number of sockets. *Third*, TM has a significantly higher throughput in both systems on four sockets than on a single socket. This is because TM has high resource demands on both CPU and memory bandwidth.

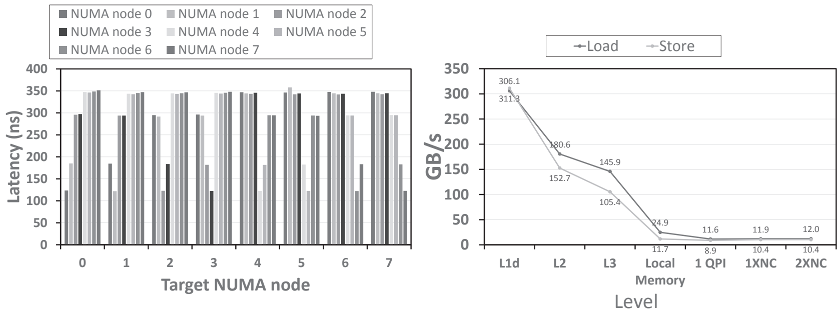
3.2 Towards Efficient Stream Processing on Multi-core Processors

In this section, we show our initial evaluation results on the effectiveness of RLS for stream processing under the NUMA effect. Overall, there are two groups of experiments. Firstly, we demonstrate the NUMA effect in our testing sever. Secondly, we show the end-to-end performance comparison.



(**a**) Evaluation of seven applications on a single socket.
(**b**) Storm with varying number of cores/sockets.
(**c**) Flink with varying number of cores/sockets.

Figure 1: Performance evaluation results on Storm and Flink



(**a**) Latency.
(**b**) Bandwidth.

Figure 2: NUMA Effect Evaluation

NUMA Effect Evaluation. Figure 2a shows how the idle latency changes varying source and target NUMA nodes. Figure 2b shows the peak bandwidth on different layers of cache and memory subsystems. From both figures, we observe a large performance difference between local access and remote (i.e., cross sockets) access.

Performance Comparison. We take word-count as an example to show the performance improvement from NUMA-aware placement (short term as NAP), and NUMA-aware placement with routing (short term as NAPR) comparing to native execution without optimization applied. Figure 3 shows the throughput improvement comparison. When source speed is slow, the performance with or without applying optimization techniques are not differentiable as the data processors are idle most of the time during execution. The benefits of applying NUMA-aware placement and routing become obvious with larger source speed, and this benefit increases along with increasing input workload.

4 Next Step

From the performance comparison study on word-count, we have obtained a promising performance improvement with relative location aware placement and routing. Our immediate next step is to obtain the performance improvement measurement

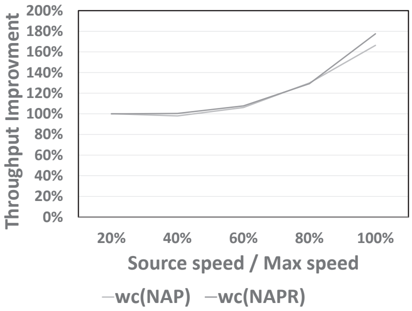


Figure 3: Speedup Evaluation based on Word-Count (WC)

of all the rest applications (in total seven). Meanwhile, there are still large space to further improve the current optimization techniques including both cost model accuracy and optimization algorithms. Hence, we are continuously enhancing the optimization techniques with new experimental results.

We have further identified a few interesting points to explore. We summarize some of them in the following discussion.

First, based on our current findings, the throughput of some applications are NUMA-nonsensitive, but the resource consumption does. Those applications have similar throughput under varying scheduling plans, but shows significantly different resource consumptions. This indicates great opportunities for multi-applications (including both NUMA-sensitive and NUMA-nonsensitive) optimization on such scale-up architectures.

Second, the current optimization designs are aimed at increasing throughput of an application, the optimization may or may not require a different design if the aim is to reduce process latency.

References

[1] D. J. Abadi, Y. Ahmad, M. Balazinska, U. Çetintemel, M. Cherniack, J. Hwang, W. Lindner, A. Maskey, A. Rasin, E. Ryvkina, N. Tatbul, Y. Xing, and S. Zdonik. "The design of the Borealis stream processing engine". In: *2nd Biennial Conference on Innovative Data Systems Research, CIDR 2005*. 2005, pages 277–289.

[2] D. J. Abadi, D. Carney, U. Çetintemel, M. Cherniack, C. Convey, S. Lee, M. Stonebraker, N. Tatbul, and S. Zdonik. "Aurora: a new model and architecture for data stream management". In: *The VLDB Journal The International Journal on Very Large Data Bases* 12.2 (Aug. 2003), pages 120–139. ISSN: 0949-877X. DOI: 10.1007/s00778-003-0095-z.

[3] L. Aniello, R. Baldoni, and L. Querzoni. "Adaptive online scheduling in storm". In: *Proceedings of the 7th ACM international conference on Distributed event-based systems - DEBS '13* (2013). DOI: 10.1145/2488222.2488267.

[4] *Apache flink*. URL: https://flink.apache.org/.

[5] *Apache samza*. URL: http://samza.apache.org/.

[6] *Apache storm*. URL: http://storm.apache.org/.

[7] A. Arasu, M. Cherniack, E. Galvez, D. Maier, A. S. Maskey, E. Ryvkina, M. Stonebraker, and R. Tibbetts. "Linear Road: A Stream Data Management Benchmark". In: *Proceedings of the Thirtieth International Conference on Very Large Data Bases*. Volume 30. VLDB '04. Toronto, Canada: VLDB Endowment, 2004, pages 480–491. ISBN: 0-12-088469-0.

[8] G. Bianchi, N. Nico d'Heureuse, and S. Niccolini. "On-demand time-decaying bloom filters for telemarketer detection". In: *ACM SIGCOMM Computer Communication Review* 41.5 (Oct. 2011), page 5. ISSN: 0146-4833. DOI: 10.1145/2043165.2043167.

[9] D. Carney, U. Çetintemel, A. Rasin, S. Zdonik, M. Cherniack, and M. Stonebraker. "Operator Scheduling in a Data Stream Manager". In: *Proceedings of the 29th International Conference on Very Large Data Bases*. Volume 29. VLDB '03. Berlin, Germany: VLDB Endowment, 2003, pages 838–849. ISBN: 0-12-722442-4.

[10] B. Chandramouli, J. Goldstein, R. Barga, M. Riedewald, and I. Santos. "Accurate latency estimation in a distributed event processing system". In: *2011 IEEE 27th International Conference on Data Engineering* (Apr. 2011). DOI: 10.1109/icde.2011.5767926.

[11] *Data request to 98 world cup web site*. URL: http://ita.ee.lbl.gov/html/contrib/WorldCup.html.

[12] Z. Fang, C. Ma, X. Wang, and J. Qu. "Mining Popular Mobility Patterns from User GPS Trajectories". In: *2016 9th International Conference on Service Science (ICSS)*. Oct. 2016, pages 180–181. DOI: 10.1109/ICSS.2016.33.

[13] *Fraud-detection*. Oct. 21, 2013. URL: https://pkghosh.wordpress.com/2013/10/21/real-time-fraud-detection-with-sequence-mining/.

[14] J. Ghaderi, S. Shakkottai, and R. Srikant. "Scheduling Storms and Streams in the Cloud". In: *ACM Transactions on Modeling and Performance Evaluation of Computing Systems* 1.4 (Aug. 2016), pages 1–28. ISSN: 2376-3639. DOI: 10.1145/2904080.

[15] J. Giceva, G. Alonso, T. Roscoe, and T. Harris. "Deployment of query plans on multicores". In: *Proceedings of the VLDB Endowment* 8.3 (Nov. 2014), pages 233–244. ISSN: 2150-8097. DOI: 10.14778/2735508.2735513.

[16] J. Gray. *Benchmark Handbook: For Database and Transaction Processing Systems.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992. ISBN: 1-55860-159-7.

[17] *Intel lab data*. URL: http://db.csail.mit.edu/labdata/labdata.html.

[18] N. Jain, L. Amini, H. Andrade, R. King, Y. Park, P. Selo, and C. Venkatramani. "Design, implementation, and evaluation of the linear road bnchmark on the stream processing core". In: *Proceedings of the 2006 ACM SIGMOD international conference on Management of data - SIGMOD '06* (2006). DOI: 10.1145/1142473.1142522.

[19] V. Leis, P. Boncz, A. Kemper, and T. Neumann. "Morsel-driven parallelism". In: *Proceedings of the 2014 ACM SIGMOD international conference on Management of data - SIGMOD '14* (2014). DOI: 10.1145/2588555.2610507.

[20] Y. Li, I. Pandis, R. Mueller, V. Raman, and G. M. Lohman. "NUMA-aware algorithms: the case of data shuffling." In: *CIDR*. 2013.

[21] Y. Liu and B. Plale. "Multi-model Based Optimization for Stream Query Processing." In: *SEKE*. 2006, pages 150–155.

[22] B. Peng, M. Hosseini, Z. Hong, R. Farivar, and R. Campbell. "R-Storm". In: *Proceedings of the 16th Annual Middleware Conference on - Middleware '15* (2015). DOI: 10.1145/2814576.2814808.

[23] T. Risch. *Uppsala University Linear Road Implementations*. URL: http://www.it.uu.se/research/group/udbl/lr.html.

[24] S. Rizou, F. Durr, and K. Rothermel. "Solving the Multi-Operator Placement Problem in Large-Scale Operator Networks". In: *2010 Proceedings of 19th International Conference on Computer Communications and Networks*. Aug. 2010, pages 1–6. DOI: 10.1109/ICCCN.2010.5560127.

[25] M. J. Sax and M. Castellanos. *Building a transparent batching layer for storm*. Technical report HPL-2013-69. HP Labs, 2014.

[26] *SGI UVTM 300H System Specifications*. URL: https://www.sgi.com/pdfs/4559.pdf.

[27] S. D. Viglas and J. F. Naughton. "Rate-based query optimization for streaming information sources". In: *Proceedings of the 2002 ACM SIGMOD international conference on Management of data - SIGMOD '02* (2002). DOI: 10.1145/564691.564697.

[28] J. Xu, Z. Chen, J. Tang, and S. Su. "T-Storm: Traffic-Aware Online Scheduling in Storm". In: *2014 IEEE 34th International Conference on Distributed Computing Systems*. June 2014, pages 535–544. DOI: 10.1109/ICDCS.2014.61.

[29] S. Zhang, B. He, D. Dahlmeier, A. C. Zhou, and T. Heinze. "Revisiting the Design of Data Stream Processing Systems on Multi-Core Processors". In: *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*. Apr. 2017, pages 659–670. DOI: 10.1109/ICDE.2017.119.

Facilitating policy adherence support in OpenStack

Max Plauth, Felix Eberhardt, and Andreas Polze

Hasso Plattner Institute, Potsdam, Germany
{firstname.lastname}@hpi.de

As a part of our efforts in the *Scalable and Secure Infrastructures for Cloud Operations* (SSICLOPS) project, our work during the *Fall 2016* period was focused on implementing a prototypical approach that facilitates policy adherence support in OpenStack with minimally invasive changes. With just a few lines of code, developers may add adherence support for policy attributes, even if the policy affects multiple OpenStack services.

1 Introduction

OpenStack[1] is an open source project that provides the tools for hosting cloud services corresponding to the *Infrastructure as a Service* (IaaS) model. A minimal OpenStack installation consists of a compute service for hosting *Virtual Machines* (VMs), as well as services for providing networking and storage infrastructures. Last but not least, an authentication service is required to complete the setup. However, a plethora of additional services is available for extending the functionality of an OpenStack setup [10]. OpenStack enjoys great popularity both in the industry as well as in the academic community [11]. In the context of the SSICLOPS project, all project partners agreed to use OpenStack as the foundation for researching cloud federation strategies due to its relevance, its open source character, and the vivid community.

With OpenStack providing the common foundation for all use case scenarios evaluated within the scope of SSICLOPS, this section documents our efforts of integrating support for the *Compact Privacy Policy Language* (CPPL) [3] within OpenStack. With policy support in place, users of OpenStack gain a fine grained tool for dictating privacy-related terms towards OpenStack instances. In the following sections, we provide an overview of prior policy concepts in OpenStack. Afterwards, we discuss the major design decisions that influenced our implementation strategy. Finally, we demonstrate with two examples how support for policy attributes can be easily implemented without having to perform extensive changes on the OpenStack code base.

[1] https://www.openstack.org/ (last accessed 2017-01-01).

2 Related Work

Here, we provide a brief overview of approaches for supporting policies in OpenStack that existed prior to our work.

2.1 oslo.policy

Even though OpenStack is comprised of many distinct projects with strictly separated concerns, certain projects may or sometimes even must use common facilities. To provide a centralized point of contact, the *Oslo* project provides a library for managing inter-project communication. Among others, the *Oslo* project provides unified interfaces for accessing databases and message queues as well as libraries for caching, logging and configuration storage.

Additionally, the *Oslo* project incorporates the package *oslo.policy*, which defines a format for specifying rules and policies and provides a corresponding policy execution engine. However, this policy engine does not suffice the requirements of (federated) clouds, as *oslo.policy* is mainly intended to be used for authorization purposes. Potentially, *oslo.policy* might be used to guard other request properties to which a yes or no answer would be sufficient, i.e., is the user allowed to instantiate a *Virtual Machine* in the United States of America. However, in the context of *SSICLOPS*, more complex policies ought to be supported [1, 2].

2.2 Swift Storage Policies

Swift is the OpenStack service for object storage. Policy support is provided at the fundamental level, as the containers holding objects can be annotated with policies such as replication rate. Policies are defined by users during the creation of a container, however, policies are restricted to core concerns of the object storage and may not be used by other OpenStack services.

2.3 Policy-based Scheduling in Nova

Upon creation of a new virtual machine, OpenStack has to decide on which host it should be instantiated. The Nova scheduler therefore attempts to locate a host that fulfills several predefined policies, including: [9]

- A sufficient amount of main memory must be available to start the VM.
- In case a specific hypervisor has been requested, it must be available on the target host.
- Only hosts that belong to the user-specified availability zone may be used to host a VM.
- The quota of the user must be considered.

Using this policy-based scheduling approach employed by the Nova scheduler enables restricting VM instantiation requests to certain hosts. [4] Custom policies can be added by OpenStack instance operators by extending the policy set employed by the Nova scheduler. Using this mechanism, it would be easy to implement a policy that makes sure that VMs of specified users are transparently instantiated in a certain region only.

The major disadvantage of this approach is that users can neither review nor edit the policies that are applied to requests. Only OpenStack operators are able to add, review or edit policies. Another shortcoming of this approach is that the policy support is restricted to the Nova component.

2.4 Group Based Policies in Neutron

The OpenStack networking component *Neutron* employs the concept of *Group Based Policies*. In the context of networking, policies can be used to specify the treatment of packets based on certain properties (e.g. the employed protocol or port). However, due to performance reasons, policies are transformed into virtual networks (including switches, router and firewalls) that satisfy the requirements rather than using a per-packet enforcement level. As this concept is very specific to the networking use case, it cannot be re-used in other OpenStack components in order to facilitate policy support.

2.5 Congress

Swift Storage Policies, Policy-based Scheduling and Group Based Policies are all internal mechanisms for various OpenStack components for evaluating policies. However, all approaches have in common that they cannot be used by other components and that they are specifically tailored to the respective domains. The *Congress*-project is a dedicated OpenStack service that aims at providing a centralized policy component for enabling compliance in cloud-based environments. As a consequence, all preceding approaches for supporting policies in OpenStack could be implemented using *Congress*. [12, 13]

Congress uses a monitoring approach in order to maintain a high degree of independence among OpenStack services. It detects policy violations in a passive mode of operation by querying the state of all involved OpenStack services in regular intervals using their corresponding APIs. In case the state of an OpenStack service deviates from a policy, the violation is logged and notifications can be triggered if configured.

One major limitation of *Congress* is that its monitoring-based approach impedes the implementation of actual enforcement mechanisms. Policies may specify how violations should be treated. In addition to logging violations and triggering notifications, policies can also be configured to revert policy violations. However, several actions are hard to revert, especially in cases where the violating action triggers many side-effects that have to be reverted as well.

Policies in *Congress* are expressed using the declarative *Datalog* policy language, which is comprised of a subset of the *Prolog* programming language. The *Congress*

API is intended to be used by OpenStack instance operator. However, using *oslo.policy*, the API can be made available for users.

In order to quantify the maturity of OpenStack projects, the OpenStack Foundation employs a maturity scale ranging from 1 to 8. After 2 years of development, *Congress* earned the lowest score 1 [6] for the Mitaka release (April 2016).

3 Design Decisions

In order to provide a better understanding of the design we came up with for our policy integration approach in OpenStack, we provide a brief discussion of some of the most crucial aspects that strongly influenced the design of our approach.

3.1 Monitoring versus Proactive Adherence

As elaborated in the context of *Congress* (see Section 2.5), proactively adherence to policies requires numerous changes in the OpenStack code base compared to a monitoring-based approach. However, the proactive approach never allows policy violations to occur in the first place, whereas monitoring-based approaches are limited to reacting on policy violations by means of logging and issuing counteracting actions. Here, we decided to aim for the proactive approach.

3.2 Versatility of Policies

The SSICLOPS policy language *CPPL* [3] supports a wide range of policy attributes. Due to this versatility of *CPPL*, policies might affect an arbitrary amount of OpenStack services. In order to deal with this high degree of versatility, each OpenStack service had to be adapted in order to support *CPPL*.

3.3 Development Process of OpenStack

OpenStack services are strictly separated in order to prevent inter-service dependencies. This level of isolation is also reflected by the development process: Services may only interact using their regular, public APIs and twice per year during the OpenStack Summits, developers meet to discuss and plan the implementation efforts 6 months ahead. [7] Contributing code to OpenStack projects involves a very complex workflow, which goes far beyond the usual habits of the typical fork-pull workflow applied on many GitHub projects. The OpenStack sources on GitHub are merely a mirror, whereas the actual sources are maintained on an OpenStack-specific Git-server[2], which held roughly 1600 repositories by November 2016.

To contribute code to OpenStack, the following process has to be adhered to: [5, 8]

- You need to have an account on the platform https://launchpad.net/.

[2] https://git.openstack.org/cgit/ (last accessed 2017-01-01).

- You need to be a member of the OpenStack Foundation.
- You have to accept the license agreement.
- The feature to be implemented has to be discussed based on a specification. Several projects are using dedicated repositories for keeping track of specifications.
- A so-called *Blueprint* is created on *Launchpad*, pointing to the specification of the feature. The *Blueprint* is used to track the progress of the feature.
- The feature is implemented in a dedicated branch.
- All code is tested extensively before it is adopted in the main branch. For the purpose of testing, the reviewing system *Gerrit*[3] is used. All changes have to be tested using automated tests. Furthermore, the changes have to be accepted manually by humans.

As this process introduces a fair amount of complexity, it would not be feasible to perform changes on the many OpenStack projects, as it would be required even by simple policies. Therefore, a central requirement for us was to keep the overhead for implementing policies as low as possible. This decision strongly influenced the design of our *policyextension*-framework, which is outlined in Section 4.

4 Integrating Policy Evaluation into OpenStack Components

One of the fundamental hurdles towards integrating proactive policy evaluation into OpenStack is the tremendous implementation effort, as all services affected by a policy have to be altered significantly.

Our implementation strategy, the *policyextension*-framework aims at providing the following characteristics:

- Interpretation and evaluation of policies should be implemented in one single location, rather than being spread across the code base of numerous OpenStack services.
- Integrating policy support should be minimally invasive regarding code changes in existing services.
- Easy maintainability of policy support code.
- Facilities should be easily extensible in order to support additional policy attributes.

[3] https://review.openstack.org/ (last accessed 2017-01-01).

To realize these goals, our *policyextension*-framework uses *PolicyExtensions* in order to integrate the evaluation of policy attributes. *PolicyExtensions* share many characteristics with plug-ins, as they are not part of the original code base. In contrast to plug-ins however, *PolicyExtensions* do not rely on plug-in mechanisms but inject their code at the locations of their own choice. The infrastructure for injecting *PolicyExtensions* is provided by the *policyextension*-framework, which also defines a certain format that *PolicyExtensions* have to adhere to by inheriting from a specific base class.

In this document, we are going to skip over the implementation details of the *policyextension*-framework itself, however we demonstrate its capabilities by presenting two examples for valid *PolicyExtensions*.

4.1 Policy Example 1: Disk Encryption

To evaluate a policy, it has to be interpreted first. The *policyextension*-framework uses the dictionary data type `dict` in *Python* to express policies in the form of key-value pairs. Our implementation of CPPL can be instructed to output such a list as result of the matching process. The following listing shows an example output for a policy that formulates the use of disk encryption:

```
{
  "storage": {
    "encryption": True
  }
}
```

To support such a policy, we adapted the *Cinder* service of OpenStack. In OpenStack, the Cinder service is responsible for providing *Block Storage*, which is indicated by the `storage` key. The embedded key `encryption` with the corresponding boolean value true then specifies, that newly created volumes must use encryption. Below, the example code demonstrates how the proactive policy adherence can be implemented by creating a new *PolicyExtension*:

```
from policyextension import PolicyExtensionBase, PolicyViolation
from cinder.volume import volume_types

class CinderEncryptedVolumeTypeRequiredExtension(PolicyExtensionBase):
 func_paths = ['cinder.volume.api.API.create']

 def create(self, func_args, policy):
  try:
   if policy['storage']['encryption']:
    volume_type = func_args['volume_type'] or volume_types.get_default_volume_type()
    if not volume_type or not volume_types.is_encrypted(func_args['context'], volume_type['id']):
     msg = "Your policy requires using an encrypted volume type."
     raise PolicyViolation(msg)
  except KeyError:
   pass
```

4.2 Policy Example 2: Restriction on Availability Zones

As a second example, we demonstrate the code for supporting a policy that restricts the instantiation of VMs to a set of whitelisted availability zones. Here, we are using the OpenStack mechanism of availability zones in order to model geographic locations:

```
from policyextension import PolicyExtensionBase, PolicyViolation
import random

class AvailabilityZoneRestrictionExtension(PolicyExtensionBase):
 func_paths = ['nova.compute.api.API.create']

 def create(self, func_args, policy):
  availability_zone = func_args['availability_zone']
  try:
   az_whitelist = policy['availability_zones']
   if availability_zone:
    if availability_zone not in az_whitelist:
     msg = ("Your policy does not allow the availability zone you selected.")
     raise PolicyViolation(msg)
   elif az_whitelist:
    func_args['availability_zone'] = random.choice(az_whitelist)
  except KeyError:
   pass
```

With these examples, we conclude the presentation of our approach for integrating proactive policy support within OpenStack. The core contribution of the *policyextension*-framework is that it grants the infrastructure for implementing support for various policy attributes with minimal effort and in a centralized component, even in cases where supporting policy attributes may involve multiple OpenStack services.

5 Outlook

Our goal for the upcoming *Spring 2017* period of the Future SOC Lab is to evaluate the policy adherence mechanisms in a federated OpenStack testbed. In addition to evaluating policy support on the level of OpenStack, we are also planning to integrate policy support in the entire application stack. As an exemplary application, we are planning to use Hyrise-R.

Hyrise-R(epilication) is a scale-out extension for the in-memory research database Hyrise. A Hyrise-R cluster consists of a query dispatcher, a single Hyrise master instance, and an arbitrary number of replicas. Users submit their database requests to a query dispatcher, which acts as a load balancer for reading queries.

Figure 1 shows two Hyrise-R clusters in a cloud environment. Cluster *one* comprises the Hyrise instances 1A, 1B, 1C. The second Hyrise-R cluster consists of the Hyrise instances 2A and 2B. The dispatchers are not illustrated and may be deployed in- or outside of the cloud environment. The Hyrise instances 1A and 2A act as masters. A number of policy attributes can be used to describe database systems, and as those the two Hyrise-R clusters.

References

[1] F. Eberhardt, J. Hiller, S. Klauck, M. Plauth, A. Polze, and K. Wehrle. *D2.2: Design of Inter-Cloud Security Policies, Architecture, and Annotations for Data Storage*. Technical report. Jan. 2016.

[2] F. Eberhardt, M. Plauth, A. Polze, S. Klauck, M. Uflacker, J. Hiller, O. Hohlfeld, and K. Wehrle. *D2.1: Report on Body of Knowledge in Secure Cloud Data Storage*. Technical report. June 2015.

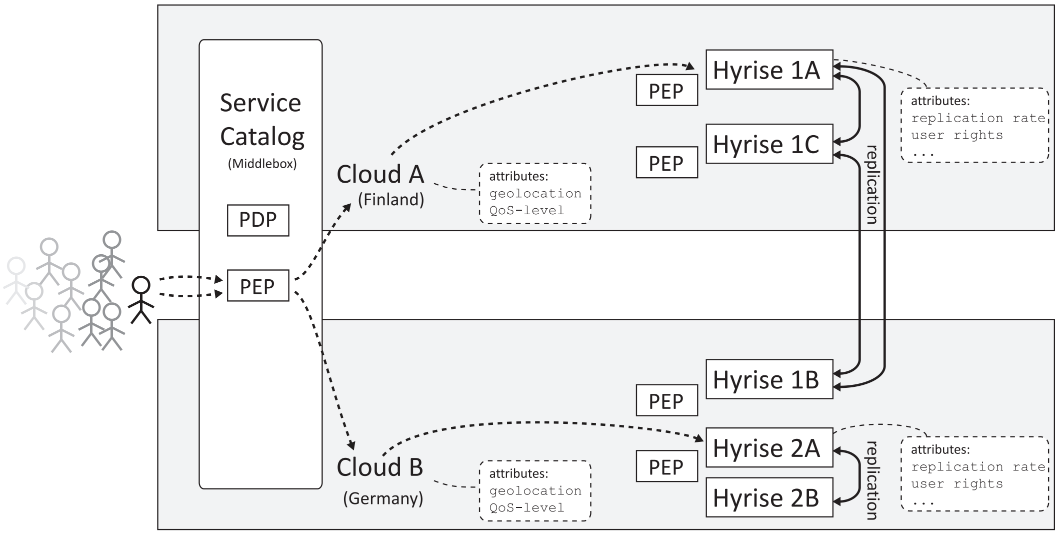


Figure 1: Use case scenario: Users request instances of the *Hyrise-R* in-memory database and annotate their requests with certain policy demands. The *policy decision point* (PDP) acts as the initial entry point and routes requests through a series of *policy enforcement points* (PEP) to process the requests accordingly.

[3] M. Henze, J. Hiller, S. Schmerling, J. H. Ziegeldorf, and K. Wehrle. "CPPL: Compact Privacy Policy Language". In: *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society*. WPES '16. Vienna, Austria: ACM, 2016, pages 99–110. ISBN: 978-1-4503-4569-9. DOI: 10.1145/2994620.2994627.

[4] Khanh-Toan Tran and Jérôme Gallard. *A new mechanism for nova-scheduler: Policy-based Scheduling*. 2013. URL: https://docs.google.com/document/d/1gr4Pb1ErXymxN9QXR4G_jVjLqNOg2ij9oA0JrLwMVRA/edit (last accessed 2017-03-01).

[5] OpenStack Foundation. *Blueprints*. URL: https://wiki.openstack.org/wiki/Blueprints (last accessed 2017-03-01).

[6] OpenStack Foundation. *Congress*. URL: https://www.openstack.org/software/releases/mitaka/components/congress (last accessed 2017-03-01).

[7] OpenStack Foundation. *Design Summit*. URL: https://wiki.openstack.org/wiki/Design_Summit (last accessed 2017-03-01).

[8] OpenStack Foundation. *Developer's Guide*. URL: http://docs.openstack.org/infra/manual/developers.html (last accessed 2017-03-01).

[9] OpenStack Foundation. *Nova Filter Scheduler*. URL: http://docs.openstack.org/developer/nova/filter_scheduler.html (last accessed 2017-03-01).

[10] OpenStack Foundation. *OpenStack Services*. URL: https://www.openstack.org/software/project-navigator (last accessed 2017-03-01).

[11] OpenStack Foundation. *OpenStack User Stories*. URL: https://www.openstack.org/user-stories/ (last accessed 2017-03-01).

[12] OpenStack Foundation. *Policy as a service (“Congress”)*. URL: https://wiki.openstack.org/wiki/Congress (last accessed 2017-03-01).

[13] OpenStack Foundation. *Welcome to Congress!* URL: http://docs.openstack.org/developer/congress/ (last accessed 2017-03-01).

Distributed Knowledge Graph Processing in SANSA

Jens Lehmann[1,2], Gezim Sejdiu[1], and Hajira Jabeen[1]

[1] University of Bonn, Germany
{jens.lehmann,sejdiu,jabeen}@cs.uni-bonn.de
[2] Fraunhofer IAIS, Bonn, Germany
jens.lehmann@iais.fraunhofer.de

Over the past decade, vast amounts of machine-readable structured information has become available through the automation of research processes as well as the increasing popularity of knowledge graphs and semantic technologies. A major and yet unsolved challenge that research faces today is to perform scalable analysis of large scale knowledge graphs in order to facilitate applications like link prediction, knowledge base completion and question answering. Most machine learning approaches, which scale horizontally (i.e. can be executed in a distributed environment) work on simpler feature vector based input rather than more expressive knowledge structures. On the other hand, the learning methods which exploit the expressive structures, e.g. Statistical Relational Learning and Inductive Logic Programming approaches, usually do not scale well to very large knowledge bases owing to their working complexity. This paper describes the ongoing project Semantic Analytics Stack (SANSA) which aims to bridge this research gap by creating an out of the box library for scalable, in-memory, structured learning.

1 Introduction

One of the key features of Big Data is its complexity. We can define complexity in different ways. It could be that data is coming from different sources, it could be the same data source representing different aspects of a resource, it could be different data sources representing the same property; this difference in representation, structure, or association makes it difficult to introduce common methodologies or algorithms to learn and predict from different types of data. The state of the art to handle this ambiguity and complexity of data is its representation or modelling in the form of Linked RDF Data.

The Linked Data follows a set of standards for the integration of data and information in addition to searching and querying it. To create linked data, the information represented in unstructured form or referring to other structured or semi-structured representation is mapped to the RDF data model, this process is called extraction. RDF has a very flexible data model comprised of triples (subject, predicate, object), that can be interpreted as a labelled directed graph (s, p, o) with s and o being arbitrary resources (vertices) and p being the property (edge from s to o) among these two resources. Thus, a set of RDF triples forms an inter-linkable graph whose flexibility allows to represent a large variety of highly to loosely structured datasets. RDF, which was standardized by W3C, is increasingly being adapted to model data in a variety of scenarios, partly due to the popularity of projects like linked open data and schema.org. This linked or semantically annotated data has grown steadily towards a massive scale[3]. Nevertheless, most existing solutions are limited to centralized environments only. In order to deal with the massive data being produced at scale, the existing big data frameworks like Spark and Flink offer fault tolerant, high available and scalable approaches to process this data efficiently. These frameworks have matured over the recent years and offer a proven and reliable method for processing of large scale unstructured data.

In the past few years, MapReduce based, and related frameworks for Big Data processing have been explored for distributed processing of RDF Data. Some examples include the Spark-based S2RDF [7] which rewrites SPARQL queries to SQL by using prior research by the RDB2RDF community and augments this approach by using precomputed semi-join tables. Approaches like SparkRDF [8], H2RDF [5] and H2RDF+ [6] use triple dataset statistics to find best merge-join orders for efficient querying. Cichlid [3] is a distributed reasoning engine, built for RDFS and OWL Horst rule-sets using the Apache Spark framework. It offers transitive closure computation, equivalent relation computation, and join processing, and proves to be 10 times faster than WebPIE due to in-memory computation. Distributed ML is primarily done through two approaches: (i) *data parallelism*, where the data is distributed into different chunks with local learners being trained on each machine; the final model is updated at specific intervals, and (ii) *model parallelism*, where the model itself and its parameters are distributed over multiple machines and trained separately on different distributed data chunks. The main motivations behind using distributed computing are being able to handle data that does not fit on a single machine, and achieve a speed-up and scalability. Systems like *Apache Spark* employ the Bulk Synchronous Parallel (BSP) synchronisation approach, i.e. each parallel iteration/task has to wait for a synchronisation step - all *sub-tasks* must finish. This ensures correctness and fault tolerance. However Machine learning applications are usually iteratively convergent in nature and this synchronisation barrier at the end of each iteration overshadows the speed-up gained by distributed computation [4]. Moreover, most of the machine learning algorithms contain interdependent parameters e.g. adaptive convergence rate of modelling parameters. This requires structure aware parallelization techniques. SANSA aims to exploit the existing communication, synchronisation and distribution techniques to optimise the performance of Distributed Structured Machine learning algorithms for large Scale Knowledge Bases.

We have decided to explore the use of these two prominent frameworks for RDF data processing.

In this paper, we introduce SANSA[4], a ***processing data flow engine*** that provides data distribution, communication, and fault tolerance for distributed computations over RDF large-scale datasets, that can process massive RDF data at scale and per-

[3] http://lodstats.aksw.org/ (last accessed 2017-01-01).

[4] http://sansa-stack.net/ (last accessed 2017-01-01).

form learning and prediction for the data. It comes with: (i) novel serialization mechanism and partitioning schema for RDF based on different strategies (i.e. vertical partitioning, unified property table and table-wise). (ii) a novel and efficient runtime querying engine for large RDF data by exploring different representation formats of distributed frameworks, namely graphs, tables and tensors. (iii) a very adaptive rule engine, which uses a given set of rules and derives an efficient execution and evaluation plan from such inference rules. (iv) out-of-the-box machine learning algorithms that work with the structured data in a distributed, fault tolerant and resilient fashion by providing analytics for gaining insights of the data for relevant trends, predictions or detection of anomalies. (v) last, but not least, a whole framework which aims to combine distributed in-memory computation framework and Semantic Technologies.

2 SANSA Framework

We now give an overview of SANSA framework. Figure 1 shows the overall architecture of SANSA that consists of four layers: *Knowledge Distribution & Representation Layer*, *Query Layer*, *Inference Layer* and *Machine Learning Layer*. In the following, we explain the role of each layer.

Knowledge Distribution & Representation Layer: It is the lowest layer on top of the existing distributed frameworks (Spark or Flink). This layer mainly provides the facility to read and write native RDF or OWL data from HDFS or a local drive and represent it in the native distributed data structures of the frameworks.

In addition, we also require a dedicated serialization mechanism for faster I/O. We aim to support Jena and OWL API interfaces for processing RDF and OWL data, respectively. This particularly targets usability, as many users are already familiar with the corresponding libraries and thus would require less time to get productive with the SANSA stack.

Query Layer: Querying an RDF graph is a major source of information extraction and searching from the underlying linked data. This is essential to browse, search and explore the structured information available in a fast and user friendly manner. SPARQL[5], also known as RDF query language, is the W3C standard for querying RDF graphs. It is very expressive and allows to extract complex relationships using intelligent and comprehensive SPARQL queries. SPARQL takes the description in the form of a query and returns that information in the form of a set of bindings or an RDF graph.

In order to efficiently answer runtime queries for large RDF data, we are exploring different representation formats of distributed frameworks, namely graphs, tables and tensors. Our aim is to have cross representational transformations and partition-

[5] https://www.w3.org/TR/rdf-sparql-query/ (last accessed 2017-01-01).

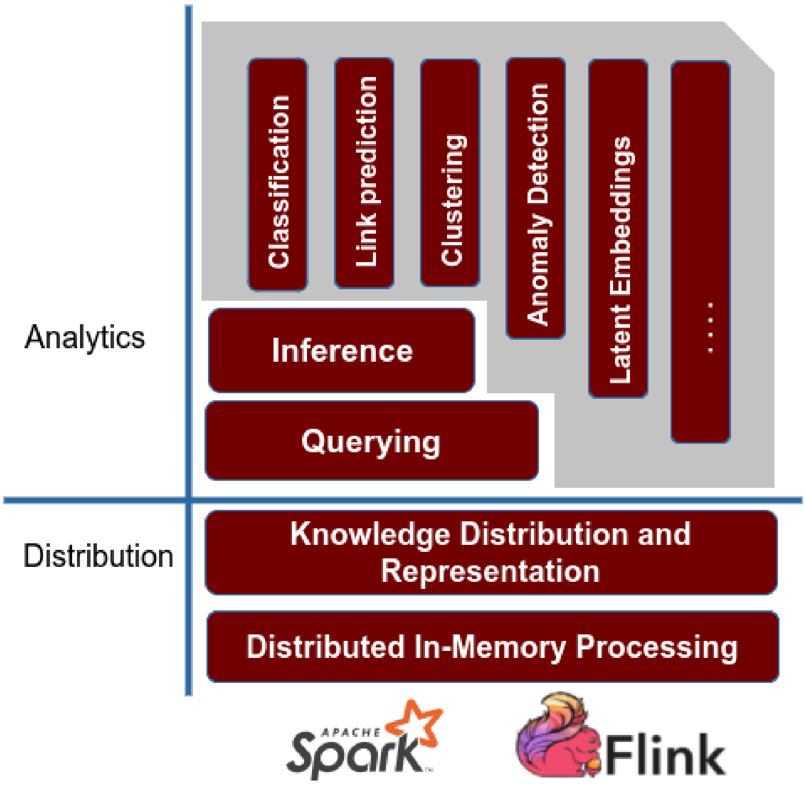


Figure 1: Overview of the SANSA stack

ing strategies for efficient query answering. We are investigating the performance of different data structures and different partitioning strategies and analyse the representations that suit particular type of queries and workflows.

SANSA contains methods to perform queries directly in programs instead of writing the code corresponding to those queries (grouping, sorting, filtering etc.). It also provides a W3C standard compliant SPARQL endpoint for externally querying data that has been loaded using SANSA.

Inference Layer: Both RDFS and OWL contain schema information in addition to links between different resources. This additional information and rules allows to perform reasoning on the knowledge bases in order to infer new knowledge and expanding the existing one. The core of the inference process is to continuously apply schema related rules on the input data to infer new facts. This process is helpful for deriving new knowledge and for detecting inconsistencies in the knowledge base. It is well known that there is always a trade-off between expressiveness of a formal language and the efficiency of reasoning in that language. SANSA contains an adaptive rule engine that can use a given set of rules and derive an efficient execution plan from a given set of inference rules.

By using SANSA, applications will be able to fine tune the rules they require and – in case of scalability problems – adjust them accordingly.

Machine Learning Layer: While most machine learning algorithms are based on processing simple features, the machine learning algorithms in SANSA exploit the graph structure and semantics of the background knowledge specified using the RDF and OWL standards. In many cases, this allows to obtain either more accurate or more human-understandable results. There exist a wide range of machine learning algorithms for the structured data. However, the challenging task would be to distribute the data and to devise distributed versions of these algorithms to fully exploit the underlying frameworks. We are exploring different algorithms namely, tensor factorization, association rule mining, decision trees and clustering on structured data. The aim is to provide out-of-the-box algorithms to work with the structured data in a distributed, fault tolerant and resilient fashion. Based on those advances, we will also be able to efficiently perform analytics to gain insights of the data for relevant trends, predictions or detection of anomalies.

3 Used Future SOC Lab resources

During the project we have used HPI 1000-core cluster. The Spark 2.0.1 have been setup as a standalone cluster with these configurations: Master:1, Workers: 12, Cores in use: 960 Total, Memory in use: 11.7 TB Total and HDFS with : Nodes: 12, Configured Capacity of 236.23 GB. We implemented the SANSA using Spark-2.0.1 and all the data were stored on the same HDFS cluster. HPI FSOC Lab provided us access to a state-of-the art, 1000-core computing cluster that is used to test SANSA for our current development.

The only disadvantage regarding flexibility was the restriction to infrequent user time slots, which in our case, dealing with large data and heavy computation is not perfectly suited.

4 Findings

Currently we are re-engineering some parts of the SANSA which deal with a performance issue and for that the Future SOC Lab resources helped us to rapidly analyse the performance bottleneck with a main focus on shuffling the data between nodes. We have worked on defining the spark data structures which are adequate, when dealing with large-scale RDF datasets. We are making use of cross representational transformations for efficient query answering. Our conclusion so far is that the Spark GraphX [2] does not speed up the processing due to complex querying related to graph structure. On the other hand, an RDD [9] based representation is efficient for queries like filters or applying a User Defined Function (UDF) on specific resources. The SparkSQL [1] Data Frames have been found efficient for efficient querying from indexed tables. We are comparing the the performance and analyse which representation suits which particular type of query answering.

5 Conclusions and Next steps

SANSA is a research-work in progress, where we are exploring existing efforts towards Big RDF processing frameworks, and aim to build a generic stack, which can work with large sized Linked Data, offering fast performance in addition to working as an out-of-the-box framework for scalable and distributed semantic data analysis. The goal of SANSA is to build a semantic analytics stack, which combines the advantages of both and allows to perform distributed (1) querying, (2) inference and (3) analytics of RDF datasets. All three aspects constitute layers in an alternative vision of the Semantic Web Stack as originally created by Tim Berners-Lee, the inventor of the World Wide Web. ***Next steps*** The SANSA basic implementation is nearly stable. However, we aim to extend the existing layers with more functionalities and adding new features. We are interested in using the Future SOC resources and to be able to test the performance of adequate SANSA Layers and to work on associated benchmarks. A further use of the HPI Future SOC Lab facilities would help us tremendously in the performance evaluation SANSA by providing us resources necessary for testing in-memory data analysis algorithms. We expect to measure a significant performance improvement for our proposed analytics.

6 Acknowledgements

This work was partly supported by the grant from the European Union's Horizon 2020 research Europe flag and innovation program for the project Big Data Europe (GA no. 644564).

The project members wish to thank SAP FSOC lab team for their effort and valuable administrative support to set-up the environment for the experiments that were conducted.

References

[1] M. Armbrust, R. S. Xin, C. Lian, Y. Huai, D. Liu, J. K. Bradley, X. Meng, T. Kaftan, M. J. Franklin, A. Ghodsi, and M. Zaharia. "Spark SQL: Relational Data Processing in Spark". In: *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. SIGMOD '15. Melbourne, Victoria, Australia: ACM, 2015, pages 1383–1394. ISBN: 978-1-4503-2758-9. DOI: 10.1145/2723372.2742797.

[2] J. E. Gonzalez, R. S. Xin, A. Dave, D. Crankshaw, M. J. Franklin, and I. Stoica. "Graphx: Graph processing in a distributed dataflow framework". In: *11th*

USENIX Symposium on Operating Systems Design and Implementation (OSDI 14). 2014, pages 599–613.

[3] R. Gu, S. Wang, F. Wang, C. Yuan, and Y. Huang. "Cichlid: efficient large scale RDFS/OWL reasoning with spark". In: *Parallel and Distributed Processing Symposium (IPDPS), 2015 IEEE International*. IEEE. 2015, pages 700–709.

[4] P. Moritz, R. Nishihara, I. Stoica, and M. I. Jordan. "SparkNet: Training Deep Networks in Spark". In: *arXiv preprint arXiv:1511.06051* (2015).

[5] N. Papailiou, I. Konstantinou, D. Tsoumakos, P. Karras, and N. Koziris. "H 2 RDF+: High-performance distributed joins over large-scale RDF graphs". In: *Big Data, 2013 IEEE International Conference on*. IEEE. 2013, pages 255–263.

[6] N. Papailiou, I. Konstantinou, D. Tsoumakos, and N. Koziris. "H2RDF: adaptive query processing on RDF data in the cloud." In: *Proceedings of the 21st International Conference on World Wide Web*. ACM. 2012, pages 397–400.

[7] A. Schuetzle, M. Przyjaciel-Zablocki, S. Skilevic, and G. Lausen. "S2RDF: RDF Querying with SPARQL on Spark." In: *PVLDB* 9.10 (2016), pages 804–815.

[8] Z. Xu, W. Chen, L. Gai, and T. Wang. "Sparkrdf: In-memory distributed rdf management framework for large-scale social data". In: *International Conference on Web-Age Information Management*. Springer. 2015, pages 337–349.

[9] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica. "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing". In: *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*. USENIX Association. 2012, pages 2–2.

A Neural Network aided predictive Mining Algorithm for Subgraphs

Lorenzo Servadei, Christian Möstl, Florian Bär, André Netzeband, and
Prof. Dr. Rainer Schmidt

Munich University of Applied Sciences
{servadei,moestl,baer,netzeban,rainer.schmidt}@hm.edu

Predictive analytics has become a widely accepted and adopted method for workload and demand [4]. It refers to the application of "[...] statistical models and other empirical methods that are aimed at creating empirical predictions (as opposed to predictions that follow from theory only), as well as methods for assessing the quality of those predictions in practices (i.e., predictive power)" [8]. Drawing upon predictive algorithms, there exist various tools, so called IT infrastructure monitoring systems (IMONs) or IT service monitoring systems (SMONs), which enable decision makers of cloud infrastructures (e.g. Chief Information Officers (CIOs) or IT administrators) to determine the probability of the occurrence of incidents and problems. Examples of such systems are IBM SmartCloud Analytics, VMware vRealize Operations, HP Operations Analytics. These tools only allow for the prediction of trends in performance data, such as Quality-of-Service (QoS), but not the detection of spots prone to failure, e.g. anomalies like single point of failures and lacking redundant items necessary for ensuring Service Level Agreements (SLAs) in the infrastructure spanned by the cloud fabric [1]. To predict the occurrence of spots reducing the reliability and resilience of the cloud-infrastructures, a graph-based approach shall be pursued. The cloud-infrastructure is represented as a graph using data from a Configuration Management Database (CMDB). In this graph the occurrence of weak spots can be predicted by searching for Anomaly-Like Subgraphs (ALS). The research aims at developing a deep learning based predictive graph mining algorithm for directed and labeled subgraphs, or more precisely CMDB graphs. It therefore contributes towards answering the research question of "How to predict structure-based anomalies in cloud infrastructures". The proposed algorithm can be implemented by IMONs and SMONs to fill this application gap.

1 Introduction

In case of highly complex cloud systems, where a big amount of nodes (entities as servers, pieces of software, virtual machines, etc.) can establish different relations (edges) among each other, it is particularly important to predict failures and anomalies, in order to avoid the partial corruption of the system [3].

This task is not simple, because the anomalies can take different forms and vary in their semblance inside a network of several thousand nodes.

In this report, we want to identify a method for detecting anomalies-prone edges present in the CMDB, which can cause a wrong distribution of resources inside a cloud system and an overload of servers/clients, determining the malfunction or breakdown of an entire cloud system.

With respect to that, we developed a Machine Learning (ML) based algorithm to identify and probabilistically quantify anomaly-prone spots in a CMDB. Such an algorithm can be a very useful feature for a cloud monitoring system [3]. The advantages of this approach are the real time detection – which would greatly decrease the reaction time of the system – and the flexibility of learning from data – which would elegantly avoid hardcoded algorithm [5]. The latter in fact is constrained by strict rules and is not able to generalize to different cloud configuration structures. Machine Learning approaches are furthermore often used in very different areas, to track various types of anomalies [7] (medicine, economics, IT-Security, etc.). Rule-based approaches would greatly adapt to a particular cloud configuration, but would be highly prone to failure in a completely different system. Another advantage of a Machine Learning System is the ability to reduce and better deal with unavoidable noise generated in the labeling phase of the anomalies, where an expert or a system is supposed evaluate each subgraph as anomalous or normal [5]. In this process, a wrong human or systematic detection could miss the true label (anomalous or normal) of a small percentage of the dataset involved.

2 Project Idea

Our research has been lead through a four steps approach, where the output of previous steps is further processed towards a consistent and robust ALS detection and classification algorithm.

2.1 CMDB Constraints

In this first research approach, we have been creating a constrained dataset of subgraphs, representing a particular CMDB of a running system [2].

The analysis of the CMDB refers to particular status of time. In the creation of our dataset in fact, we did not take into account the temporal dimension, which will be further researched in the next papers.

The first step of our subgraphs generation has been the creation of the different nodes available in our CMDB.

These nodes correspond to *Server*, *RAM*, *VM*, *OS*, *Manufacturer*, *CPU*, *Licence*, *Software*, *HardDisk*, *ITService*.

The possible relations among these nodes, that is the edges of the subgraph, are instead the properties *runs on*, *in*, *produces*, *depends on* and *requires*.

Established that, it is necessary, with the purpose to further describe our environment, to determine the physical constraints of the system. The latter are the rules implicit in the domain where our graphs are generated. The constrains form directed edges, and allow us to represent the constrained domain graphically (see Figure 1).

In our experimental CMDB we have been determining following rules:

- *The Server node can be availed from each node, apart from itself, IT-Service and license.*
- *The RAM node can be availed only by the Manufacturer.*
- *VM cannot be availed.*
- *The OS node can be availed only by the Manufacturer and Software nodes.*
- *The Manufacturer cannot be availed.*
- *The CPU can be availed only by the Manufacturer Node.*
- *The License node can be availed only by the Software.*
- *Software can be availed by Manufacturer, IT-Service and itself.*
- *The HardDisk node can be availed by the Manufacturer.*
- *The IT-Service node cannot be availed.*

This configuration of constraints, is not a valid generalization, but is instead a specification of our domain. These rules are in fact depending on specificities of a cloud environment or of sub-environment of a cloud system. The reason for our choice is to be found in the central role taken by the server node in the CMDB representation.

This leads to a better server-oriented analysis of the anomalies and prediction towards server overloading or isolation.

2.2 Subgraphs Generation

Under these predefined constraints, we structured the generation of our artificial dataset.

We organized a dataset of 10 000 subgraphs, 5000 defined as normal and 5000 defined as anomalous.

In order to do so, we developed some further constraints in the adjacency matrix of the node relations, describing the status of the subgraph.

The constraints created for the generation of normal status subgraphs have been applied through the python numpy pseudo-random method *randint*, which expresses a random choice within a certain interval.

This method has been helping us to create a certain variance inside our dataset, which is very important for better learning the mapping input-output function of the dataset.

Some examples of constraints expressed by the normality status are: A single server cannot host more than two virtual machines, or each software should have at

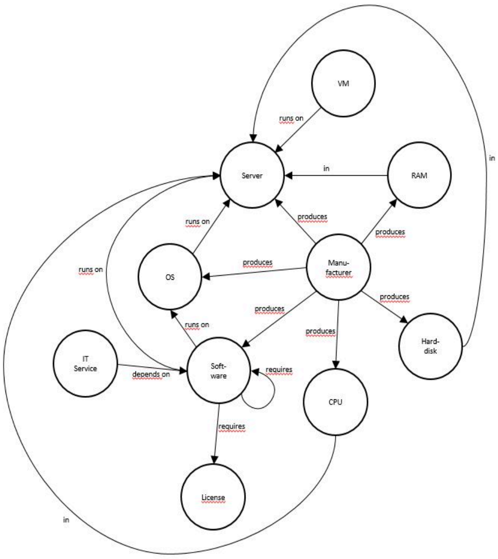


Figure 1: Node and Edges in the constrained Subgraph

least one license, or each server should have at least one RAM module, but not more than two.

These constraints, as already mentioned, are particular to our system, and follow given configuration desired by the CMDB.

At the same time, they represent the feature to learn, which are commonly used in each particular system configuration or sub-configuration.

As a dependent constraint, we add, in the normal samples generator, the complex constraints that load of edges between the server and the OS, the VMs and pieces of Software installed cannot overcome the 25 units. This particular feature expresses the maximal load from these three elements to the server node, in order to describe the subgraph configuration as normal.

A similar procedure will be involved in the creation of anomalous graphs, with the peculiarity of XOR constraints (e.g. a subgraph can be anomalous in case of no software on a Server, but also in case of too many installed).

After this process of constrained stochastic sampling, with the uniform distribution for the anomalous and the normal subgraphs generation, we are going to have a dataset which is equally divided in 5000 anomalous and 5000 normal subgraphs. This situation is optimal for detecting and gives the same importance to the relations among the nodes, and to normality and anomalous cases [6].

With the sampling, the adjacency matrix representing the connections among nodes of the subgraph will be substituted by a vector which represents, in its components, the quantity of edges connecting two particular nodes in a specific direction.

As final step of the subgraph generation, in order to have a realistic dataset and avoid a trivial classification hypothesis, we added noise to the dataset. The way Gaussian noise has been added to the dataset is proportional to the variance of a

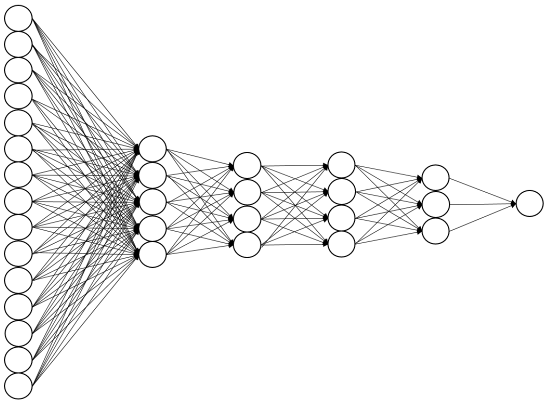


Figure 2: Architecture of the MLP Classifier

single feature (column) of our multidimensional array of generated samples, as shown in the code below:

```
varianceMatrix = np.var(oneDimRelArray[:, :15], 0)

# I multiplied the variance of each column by two, in
# order to create cases of not prediction,
# which simulate real valued examples

for idx, val in enumerate(varianceMatrix):
  oneDimRelArray[:, idx] =
    oneDimRelArray[:, idx] +
    np.random.normal(0, val*2, 9999)
```

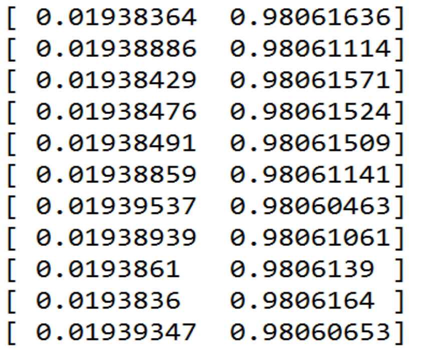
In this way, the noise added depend on the variance in the number of each particular type of edge. In order to do that, we treated the features as independent and normally distributed. The noise generated as well, follows the normal distribution.

2.3 Classification of the Subgraphs

In order to accurately learn our mapping function for this supervised task, we have been shuffling the examples and have chosen the best model for our classification.

The best performing classification architecture has been identified in a multi-layer perceptron (MLP) neural network, which is a feedforward neural network where the layers of neurons are fully connected, so that each neuron is connected to all the neurons of the subsequent layer.

We availed of four hidden layers (5, 4, 4, 3) and performed a weight update through a constant learning rate, as shown in Figure 2. The learning algorithm proposed for the learning phase has been performed by the *adam* solver, which is an optimization of the stochastic gradient descent (SGD).



```
[ 0.01938364  0.98061636]
[ 0.01938886  0.98061114]
[ 0.01938429  0.98061571]
[ 0.01938476  0.98061524]
[ 0.01938491  0.98061509]
[ 0.01938859  0.98061141]
[ 0.01939537  0.98060463]
[ 0.01938939  0.98061061]
[ 0.0193861   0.9806139 ]
[ 0.0193836   0.9806164 ]
[ 0.01939347  0.98060653]
```

Figure 3: Confidence on the binary classification of the ALS

2.4 Evaluation of the Algorithm

The accuracy obtained in the classification has been of 98.8 % in the cross-validation phase (we split our dataset in 70 % as a training set, 15 % as a cross-validation set and 15 % as a test set).

Such an accuracy, in uniformly distributed dataset, should be taken as a positive result [6].

With respect to our final question, that is the grade of anomaly-likelihood, we implemented a third set of anomaly like subgraphs. This time, we did not label them, with the intention to measure the likelihood of the dataset, that means, how much our system is confident over one class (normality) or another (anomaly). It has to be pointed out that the third set of samples (ALS) have generally an intermediate value on the constraints of anomaly and normality. If at least one of the value is in the anomaly-like constraint interval, the subgraph will be considered ALS, even if all the other values are in the normality interval.

To calculate the confidence of the model to classify our third set of samples, we require the results of the last neurons (output) of our network, where the confidence is expressed, as shown in the code.

$$P(y|X) = 1/(1 + exp(A * f(X) + B))$$

To keep the output value in the interval between 0 and 1, we used a non-linear function called *sigmoid* function, while for the rest of neurons we used, as activation function, the *tanh* non-linearity. We notice that the values expressed from the model confirm our expectation, as shown in Figure 3.

The values of the model express a high confidence (being the model very accurate in general), and confirm the fact that the third set can be classified as normal with high belief.

Nevertheless, we notice on the percentage decimal a change in confidence from sample to sample.

We have been researching on this value and we determined that the value obtained is reflecting the Euclidian distance of each subgraphs vector dimension to the nearest extreme of the anomalous graph range.

3 Used Future SOC Labs Resources

For the project we required and obtained access to 64 GB of RAM memory, 24 CPU cores and GPU access as well. Furthermore, we have been granted with the possibility to use four VMs and a Hard Disk storage of 512 GB. On this provided support, we ran our python scripts, for graph generation and for training the classification model.

4 Findings

In our research, we performed an accurate classification in our validation set (98.8 % accuracy with an equal distribution inputs) of anomalous and normal subgraphs in a noisy dataset. This leads to a fast and accurate tracking of anomaly in a cloud system CMDB.

Furthermore, we could learn, despite of noisy data, a mapping function to reveal the anomaly like subgraphs, which represents the spots of the cloud system prone to transform into anomalies. We could learn different values of confidence, which maps the proximity of the subgraph to the anomalous state. As a proof, we could measure this proximity as the result of Euclidian distance of each edge quantity to the nearest extreme of the anomalous constraint. This gives a justification and explanation to our results, and a motivation for further research.

5 Next Steps

The results obtained show the possibility to introduce a deep learning algorithm able to better identify anomalies, in situations where strict rules do not fit to the dataset. As shown, the ability of learning is adding flexibility if compared to hard coded algorithms, which appear not able to deal with noise and specificity of a configuration. In the proposed environment, we classified with high accuracy (98.8 %) in a noise augmented dataset. Furthermore, we were also able to calculate the log likelihood of our subgraphs being anomalous, and strictly relate the result to the hardcoded constraints imposed to each class. This means that the machine learning algorithm is robust and able to deal with a big variance in the features and noise, which let think to a generalized approach for the future (as a possible application to several configuration of the cloud infrastructure).

In the next steps, it will be important to implement real data out of an existing cloud infrastructure. The analysis of anomalous cases can drive us to improvements of our algorithm, and furthermore, to relate with particular configuration problems that are not present in a self-generated dataset. Another component which should be taken into account in a monitoring system, is the time dimension. In the next papers we will analyze the mutation of the anomalies within a time series perspective: This could help us to realize even more accurate algorithms for ALS detection.

References

[1] F. Bär, R. Schmidt, and M. Möhring. “Fabric-Process Patterns”. In: *Enterprise, Business-Process and Information Systems Modeling*. Springer, 2014, pages 139–153.

[2] R. J. Colville and G. Spafford. “Configuration management for virtual and cloud infrastructures”. In: *Gartner2010* (2010).

[3] S. Fu. “Performance metric selection for autonomic anomaly detection on cloud computing systems”. In: *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*. IEEE. 2011, pages 1–5.

[4] J.-J. Jheng, F.-H. Tseng, H.-C. Chao, and L.-D. Chou. “A novel VM workload prediction using Grey Forecasting model in cloud data center”. In: *Information Networking (ICOIN), 2014 International Conference on*. IEEE. 2014, pages 40–45.

[5] T. D. Lane. *Machine learning techniques for the computer security domain of anomaly detection*. MyScience, 2000.

[6] B. Pang, L. Lee, and S. Vaithyanathan. “Thumbs up?: sentiment classification using machine learning techniques”. In: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics. 2002, pages 79–86.

[7] L. Servadei, R. Schmidt, and F. Bär. “Artificial Neural Network for Supporting Medical Decision Making: A Decision Model and Notation Approach to Spondylolisthesis and Disk Hernia”. In: *On the Move to Meaningful Internet Systems: OTM 2016 Workshops*. Edited by I. Ciuciu, C. Debruyne, H. Panetto, G. Weichhart, P. Bollen, A. Fensel, and M.-E. Vidal. Cham: Springer International Publishing, 2017, pages 217–227. ISBN: 978-3-319-55961-2. DOI: 10.1007/978-3-319-55961-2_22.

[8] G. Shmueli and O. R. Koppius. “Predictive analytics in information systems research”. In: *Mis Quarterly* (2011), pages 553–572.

Text Mining on Job Offers Using SAP HANA
Analyzing Skill and Competency Requirements for Industry 4.0

Marlene Knigge, Sonja Hecht, Loina Prifti, and Helmut Krcmar

Technical University of Munich
Chair for Information Systems
{marlene.knigge,sonja.hecht,loina.prifti,krcmar}@in.tum.de

In this project, we used text analysis and text mining means provided by SAP HANA and SAP Predictive Analytics Library (PAL) on job offers in the field of Industry 4.0. Recent technological developments, e.g., in the fields of sensors, databases, or data analytics, resulted in the fourth industrial revolution – or Industry 4.0. These lead to changes in our working environment. Thus, employees are facing changing skill and competency requirements for which they have to be qualified appropriately. To find out, which requirements are demanded in an Industry 4.0 environment, we collected job offers from German online job portals for several months. We applied text analysis and text mining methods and algorithms offered by SAP HANA and SAP PAL to them in order to extract skill and competency requirements.

1 Introduction

The fourth industrial revolution – also referred to as "Industrie 4.0", or Industry 4.0 – is significantly changing the way we live and work [6, 7]. The increasing integration of the real world and the virtual reality results in disrupting solutions such as the Internet of Things (IoT), Smart Factories, Cyber Physical Systems (CPS), and the increased use of Embedded Systems – which are the core elements of "Industrie 4.0" [6, 7]. These changes will have far-reaching implications on business processes, business value creation, business models, downstream services, and work organization [7]. For example, in an "intelligent company the employees supervise the flexible manufacturing process with devices for the computer based reality perception (augmented reality devices like tablets or smart glasses), the employees immediately react if problems or process changes appear and the employees are assisted by fine sensory robot units" [5]. These technological developments challenge the employees: "The requirements for the digitized skilled work will rise because the processes are interconnected and more complex, particularly with reference to the overlap of technical, organizational and social spheres of activity and the work process in the company" [5]. Employees need to be qualified for new tasks and even new kinds of jobs [9]. acatech et al. [1] conclude in their study regarding competency development, that qualification will be one of the main factors for the successful management of the digital shift in Germany. Training and education will have to be adjusted to the new skill and competency requirements. Therefore, it is crucial to identify the skills and competencies that will become more important.

In this project, we want to identify and analyze skill and competency requirements for Industry 4.0. Our approach is to apply text analysis and text mining methods and algorithms on job offers from the area of Industry 4.0. We want to extract skill and competency requirements as well as metadata such as job titles, region, industry, etc. and to analyze connections between those. In a second step, we want to automate this process as far as possible and apply it on a regular basis in order to be able to analyze changes in skill and competency requirements over time.

2 Project Goal

Our goal is to contribute to the analysis of unstructured data, in this case, online job offers from different sources. Therefore, we want to improve the application of text analysis and text mining algorithms on text documents – in this case job offers. First, we want to extract skill and competency requirements from the job offers. For this, we want to find out which algorithms are suited best for this task. Second, we want to extract metadata, such as job title, company or region. For doing so, we will have to implement custom dictionaries and extraction rules. In our follow-up project, we want to analyze associations between skill and competency requirements and metadata. Moreover, we aim to execute our analysis on a monthly base to be able to monitor changes over time.

To achieve the first two steps in the scope of this project, we apply methods and algorithms offered by SAP HANA as well as by the SAP Predictive Analysis Library (PAL) to a data sample, which consists of manually collected job offers in the area of Industry 4.0. The results of the different algorithms applied are evaluated and compared.

3 Project Design

We build our work on a dataset which is manually collected on a monthly base. The collection is still ongoing (cf. section 3.2). We process our data on SAP HANA and use algorithms provided by SAP PAL (cf. section 3.1).

In our first subproject, we concentrate on the extraction of skill and competency requirements from job offers. Therefore, our goal is to apply different algorithms such as k-nearest neighbour (kNN), C4.5, or support vector machine (SVM) and evaluate the results.

In our second subproject, we want to extract metadata from job offers, such as information regarding the company, industry, region etc. We want to analyze this metadata to find out if there are patterns in connection to competency profiles, e.g., regarding industry, region, or company size.

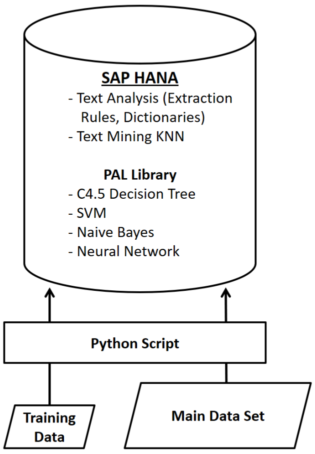


Figure 1: IT architecture (Source: own illustration)

Table 1: Dataset: Manually collected job offers

	Job Portal A	**Job Portal B**	
Nov. 2016	**178**	**100**	
Dec. 2016	**271**	250	
Jan. 2017	210	175	
Feb. 2017	234	174	Total
Total	893	699	**1,592**

3.1 System Configuration

The Hasso Platter Institut (HPI) provided us with a SAP HANA SPS12 with 1 TB RAM and 32 Cores (CPU). Additionally, we used the SAP HANA Predictive Analytics Library (PAL) and the Eclipse-based SAP HANA Studio, version 2.3.10. For uploading the job offers, we implemented a script in Python (version 3.5.2) as this was a fast and convenient solution (cf. Figure 1).

3.2 Dataset

For extracting skill and competency requirements, we collected job offers manually on a monthly base over four months (cf. Table 1, collection still ongoing). This dataset currently comprises 1,592 manually collected job offers from two different online job portals from the German online job market, found when applying the search term "Industrie 4.0". Due to the availability of the dataset during the implementation phase, the training and application of our algorithms was mainly conducted on the data from November (job portal A + B) and December (job portal A) 2016. (549 job offers).

3.3 Data Pre-processing

We started with the pre-processing on the data collected from job portal A in November and December 2016, and those from job portal B collected in November 2016 (549 job offers). All corrupt files and duplicates were removed from the dataset. A part of the job adverts was stored as html-files. For those, the html-tags were removed automatically by creating a full-text index. The other part of the job offers was stored as pdf-files. In some of these files, there was a forced page-break in a sentence. We did not try to resolve this issue because it should not have a big impact on the results with regards to the methods we apply.

4 Skill and Competency Requirements Extraction

For skill and competency requirements extraction, we tried two approaches. First, we did the extraction on sentence level. Next, we compared it with the extraction executed on document level (job offers).

For the skill and competency requirements extraction on job offers, we first used the native SAP HANA Text Analysis functionalities.

4.1 Custom Dictionaries and Extraction Rules

Based on the default ″EXTRACTION_CORE″ configuration, we created a custom configuration for text analysis. We included a custom dictionary and custom extraction rules. The custom dictionary comprises the skills and competencies we want to extract, e.g., "Eigenverantwortung" (self-responsibility). We added synonyms by using an online dictionary for German language, Duden [4]. Only terms relevant in the context of job offers were included. Our dictionary does only capture a fraction of the occurrences of the skill and competency requirements that can be found in the job offers. This is because it only matches the words stated in the dictionary with the text from the job offers. However, in job offers, skill and competency requirements are described in diverse manners. In the given example, self-responsibility could also be expressed through an adjective such as ″eigenverantwortliches Arbeiten″ (working independently). The number of possible expressions is further increased by the different declined forms of the adjective as well as by possible synonyms for the noun it describes. The dictionary only offers a limited way to capture these variations. To solve this problem, we created custom extraction rules. For our example, a rule is looking for two consecutive words, the first one having the stem of independently and the second one a stem matching a list of words, e.g., "working", ″way of thinking″, etc. Moreover, we used custom rules to capture information on work experience, language skills and requirements for expertise. Table 2 exemplarily shows the results for the skill and competency extraction for one job offer.

Table 2: Exemplary result of skill and competencies requirements extraction

Category	Requirement
Qualification	Studium: Informatik, Wirtschaftsinformatik
Work experience	—
Competencies	Technikaffinität Projektmanagement Analytische, pragmatische Denkweise Sozialkompetenz Kommunikationsfähigkeit
Language	Deutschkenntnisse

4.2 Extraction on Sentence Level

Our second approach was to split up each job offer into single sentences. Balbi [2] assumed that only certain sentences contain information concerning skill and competency requirements for jobs. Therefore, he expected the results of the information extraction to be better, when concentrating on those sentences only. Therefore, our next step was to try if we could refine our results by analyzing single sentences.

We started with the creation of a training dataset. Therefore, we extracted the sentences from 100 randomly chosen job offers of our dataset using the standard "LINGANALYSIS_FULL" full-text index provided by SAP HANA. The resulting training dataset contains 3,461 single sentences. These sentences were labelled twice by two independent researchers with regards to the fact if they contain at least one skill or competency requirement or not. 238 sentences of the training dataset were labelled as "1" for "containing a skill or competency requirement". 3,223 sentences were labelled as "0" for "not contain a skill or competency requirement".

On this training dataset, we created a term frequency–inverse document frequency (TF-IDF) vector representation of the sentences. One shortcoming of SAP HANA we were facing is, that there is no efficient way to save a term-document matrix – while other products may comprise specific storage structures to do so. So to reduce computational effort when creating the vector representation and training the algorithms, we used a subset of 983 terms. These were chosen based on the number of documents they appear in and their inverse document frequency. We trained and tested the PAL algorithms Naive Bayes, C4.5 Decision Tree, Support Vector Machine (SVM), Backpropagation Neural Network (NN) as well as the k-Nearest-Neighbor (kNN)-based classification provided by the SAP HANA Text Mining component on different subsets of this dataset. By using the NN algorithm, we achieved the best results with an overall F-score of 90 %, shortly followed by the SVM with 87 %. For an overview of our approach, cf. Figure 2.

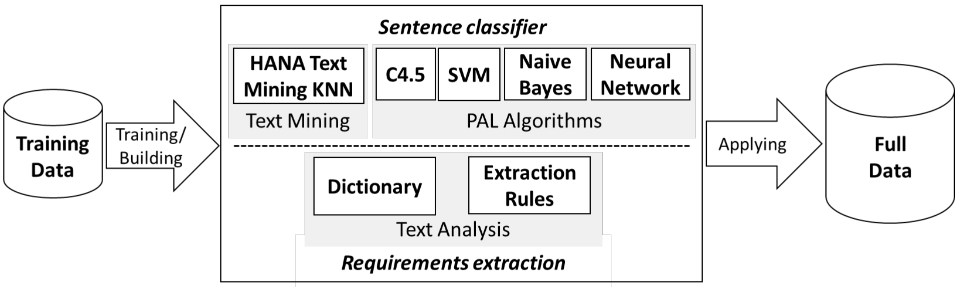


Figure 2: Skill and competency requirements extraction (Source: own illustration)

4.3 Extraction on Document Level

Second, we extracted skill and competency requirements from complete job offers (still ongoing). This approach proved to be more efficient as the extraction on sentence level as we can skip the step of extracting single sentences from the full-text-index of the job offers first. We then applied SAP HANA Text Analysis with dictionaries and extraction rules to the data. We created a second test dataset for being able to evaluate this approach as well. Therefore, 15 randomly chosen job offers were analyzed manually regarding the skill and competency requirements they contain. By comparing the extraction on sentence and on document level, we do not have reason to believe that the quality of the skill and competency requirements extraction is higher if we conduct it on the basis of single sentences. Moreover, for the extraction of metadata (cf. section 5), it is necessary to analyze complete job offers in order to be able to associate the different skill and competency requirements with the metadata (e.g., interdependencies between job title or region on the one hand, and skill and competency requirements on the other).

Next, we used the training dataset with the previous labelled sentences from our first approach (cf. section 4.2) to discover weaknesses in the extraction of requirements on document level. We investigated sentences where a requirement was found, but which were classified as containing no requirement (false positives). A main cause for the misclassification was that competencies were used to describe the company, the tasks or the job environment. Requirements were also used in hidden tags (only relevant in the case of html-files). Some cases were unclear even after the two independent manual classifications. One way to avoid technical misclassification is to incorporate the context where a requirement is used. This can be done by combining the skill and competency requirements extraction with the sentences classification and only apply it to sentences classified as containing a requirement. Another approach is to refine the results with SQL queries. E.g., only these occurrences of languages, which appear in conjunction with the word "Kenntnis" ("skill"), will be considered as skill requirement.

The next step will be to assess the results of the skill and competency requirements extraction in a qualitative way. Therefore, we will choose job offers randomly, extract the requirements manually, and compare the results with those of the automatic extraction.

5 Metadata Extraction

Additional to the skill and competency requirements, we also extracted further metadata from the job offers. We combined the requirements and the metadata in order to discover association patterns, by further analysis. The extracted metadata included information such as job title, company, region, and certain general conditions of the job offer. In addition, information about the company such as the number of employees, industry, and the structure of the company were extracted.

Our first approach was to extract this information using the entity and fact extraction functionalities provided by the standard text analysis library of SAP HANA with the configuration "EXTRACTION_ CORE_ENTERPRISE". This offers the possibility for identifying companies and organizations from the text as well as facts such as job titles, locations and addresses. In our setting, the results of company and organization recognition did not lead to useful results. Compared to manual classification of the result tables, only 60 % of the entities classified automatically as "ORGANIZATION/COMMERCIAL" really did contain companies. The recognition of job titles classified automatically as "TITLE" was even worse. Only 8 % of the actual job titles were extracted correctly with the standard configuration. One explanation is, that the skills and competencies required from a user regarding an IT product like "SAP" are often classified as corresponding enterprise by the automatic extraction. The industry, which a company belongs to, is often recognized as name of an enterprise. To bypass these issues, the creation of custom dictionaries is required. Yet, the extraction of the job location worked well in different levels of detail using the standard extraction configuration.

The dictionaries for all categories were created using different approaches. First, we manually classified the result tables of the extraction when a certain term was extracted at least three times from the training dataset of 549 job advertisements (cf. section 3.2). Second, different databases and data sources were used to create custom dictionaries for companies, job titles, and industries. For companies, we used databases such as the "German enterprises listed on the stock exchange" [8] and the "German Top 1,000 family enterprises" [3]. We are still working on including more databases for the other areas. We compare the quality of the results of these two approaches, as well as a combination of both to reach better results than with using the standard "EXTRACTION_CORE_ ENTERPRISE" configuration. Moreover, different additional information about the company, such as the number of employees, or the position in the market (e.g., "Hidden Champion") can be identified via certain keywords stored in the dictionary. With the same approach, additional information about the job type (e.g., "Full-time position") can be identified.

With the combined approach, we were able to resolve one critical matter: the recognition of IT-enterprises without confusing them with their products. Within the custom dictionaries, the custom entity type "COMPANY/CUSTOM" was created. With this approach, 351 entities of this type were extracted from the training dataset. In comparison, from the standard company extraction of the standard configuration 1,729 distinct values of the entity type "ORGANIZATION/COMMERCIAL" have been extracted. As mentioned before, the extraction of these entities is oversensitive. For 47 % of the 549 job offers analyzed with this configuration and our current dictionary, we could identify a company, even at different level (e.g., "Bosch Thermotechnik GmbH" as part of "Bosch Gruppe"). Not every analyzed job description can be linked with a company. However, for 53 % of the job offers, we could not identify a company with our current configuration. One explanation is, that some of the job offers are anonymous. We assume, that we can increase the company detection rate significantly by using larger training data set and further improvement of our dictionaries and extraction rules.

For further improvement of the quality of the described custom configuration, custom extraction rules and language dictionaries will be defined in addition to the dictionaries used. Moreover, the amount of training data will be increased. In addition, the usage of custom variant generators can be an option as usage of standard variant generation forms, which is supposed to generate predictable standard variants (e.g., "MAN SE" out of "MAN AG") does not significantly improve the results.

The quality of the tested extraction configuration was evaluated by using SAP HANA data analysis tools and different SQL queries. All important and interesting facts ordered and grouped by the ID of the analyzed job offers give an overall overview about quantity and quality of extracted information per text file. As one example, in the job advertisement with the ID 38 the "Bosch Thermotechnik GmbH" (COMPA-NY/CUSTOM) in "Stuttgart" (LOCATION) is looking for a "Softwareentwickler" (JOBTITLE/IT) as "Feste Anstellung" (POSITION/INFO) with the possibility for "Teilzeit" (POSITION/INFO).

6 Limitations

We faced trouble when creating the extraction rules. The activation of a flawed .hdbrule-object did not finish. It was not possible to cancel it, and moreover, it blocked the activation of all other objects in the system. The use of guillemets instead of quotation marks around the file location in an import statement and referencing it by the given name caused this issue. During activation no error was indicated because the check of the file never completed. It took us a lot of trial and error to locate the error.

As described in section 3.4 we faced the fact that currently there is no efficient way to save a term-document matrix in SAP HANA.

We still have to apply the text analysis and text mining methods we implemented on the total of our (still growing) dataset.

7 Conclusions and Outlook

After collecting a set of sample data and creating training and test datasets, we have been able to successfully apply text analysis and text mining on job offers.

When labelling the separated sentences using the algorithms, we had the best results regarding the recognition of skill and competency requirements with NN (90 %) and SVM (87 %). By combining SAP HANA Text analysis on complete job offers with the analysis on the basis of single sentences or SQL queries, we want to decrease the false positive rate.

Using custom dictionaries and extraction rules, we were able to extract metadata from job offers. The next step will be to analyze these results and to discover associations between meta data and skill and competency requirements.

In a follow-up project, we want to automate our whole application, starting from the extraction of the job offers, over the pre-processing, to the skill and requirement and the metadata extraction. This will allow us to do more complex analysis on our dataset and include analysis regarding changes in skill and competency requirements over time.

8 Acknowledgments

We would like to thank our master students Sebastian Hagn and Alexander Weinhart for being part of our project and for their dedication and their ideas. Moreover, we wish to thank our student assistants Leon Kevin Birkel and Tristan Hopf for their support.

References

[1] acatech – DEUTSCHE AKADEMIE DER TECHNIKWISSENSCHAFTEN in Kooperation mit Fraunhofer IML und equeo. *Kompetenzentwicklungsstudie Industrie 4.0: Erste Ergebnisse und Schlussfolgerungen*. Apr. 2016.

[2] S. Balbi and E. Di Meglio. “A Text Mining Strategy based on local contexts of words”. In: *Proceedings of the Journées internationals d'Analyse statistique des Données Textuelles (7th International Conference on the Statistical Analysis of Textual Data)*. Volume 4. Mar. 2004, pages 79–87.

[3] *Die 1000 größten Familienunternehmen 2016*. URL: https://die-deutsche-wirtschaft.de/die-liste-der-1000-groessten-familienunternehmen-in-deutschland/ (last accessed 2017-03-14).

[4] *Duden*. URL: http://www.duden.de (last accessed 2017-03-13).

[5] J. Gebhardt, A. Grimm, and L. M. Neugebauer. “Developments 4.0-Prospects on future requirements and impacts on work and vocational education”. In: *Journal of Technical Education (JOTED), Jg* 3 (2015), pages 117–133.

[6] *Industrie 4.0: Die neue Hightech Strategie – Innovationen für Deutschland*. URL: http://www.hightech-strategie.de/_dpsearch/highlight/searchresult.php?URL=http://www.hightech-strategie.de/de/Industrie-4-0-999.php&QUERY=industrie+4.0 (last accessed 2016-03-09).

[7] H. Kagermann, J. Helbig, A. Hellinger, and W. Wahlster. *Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Securing the future of German manufacturing industry; final report of the Industrie 4.0 Working Group*. Forschungsunion, 2013.

[8] Wikipedia contributors. *Liste der börsennotierten deutschen Unternehmen — Wikipedia, Die freie Enzyklopädie*. Mar. 7, 2017. URL: https://de.wikipedia.org/w/index.php?title=Liste_der_b%C3%B6rsennotierten_deutschen_Unternehmen&oldid=163376608 (last accessed 2017-03-15).

[9] *Zukunftsprojekt Industrie 4.0. Digitale Wirtschaft und Gesellschaft*. URL: https://www.bmbf.de/de/zukunftsprojekt-industrie-4-0-848.html (last accessed 2016-03-09).

Security analytics of large-scale heterogeneous data

Andrey Sapegin, David Jaeger, Feng Cheng, and Christoph Meinel

Hasso Plattner Institute, Potsdam, Germany
{firstname.lastname}@hpi.de

Modern Security Information and Event Management (SIEM) systems should deal with large volumes of data coming from heterogeneous sources from all over enterprise network. The log messages come from routers, proxy and web servers, client machines, domain controllers etc. The content of such messages could be very different and reflects different information, which should be extracted and correlated to perform an analysis. We perform such analysis under Future SOC Lab project and show how to efficiently process, correlate and analyse the heterogeneous security data.

1 Introduction

According to the recent Gartner report [4], SIEM system vendors are incorporating advanced analytics into their solutions. Since the SIEM system processes data from multiple sources and in different formats, such advanced analytics is hardly possible without data normalisation. Besides the data normalisation, SIEM systems also face other challenges during data processing, e.g. in performance and accuracy. In this report, we present a prototype of such a SIEM system, which is capable of high speed normalisation, processing, correlation and analysis of Big Security Data. We prove our concepts on the real dataset from our partner enterprise, containing hundreds of millions of log messages. To successfully analyse it, we utilised novel algorithms and approaches and executed them on the infrastructure provided by Future SOC Lab, as described in the sections below.

1.1 HPI Future SOC Lab resources

Under this project we have used various hardware and software resources provided by HPI Future SOC Lab. It includes an access to VMware ESXi hypervisor with 256 GB RAM and 64 CPU cores, as well as shared access to 2 SAP HANA [5] instances, one with 1 TB RAM and another one with 250 GB RAM.

The access to these resources helped us to perform security analytics on the dataset, which will be described below.

1.2 Dataset

We prove our concepts on the dataset from the large multinational company presented in Figure 1.

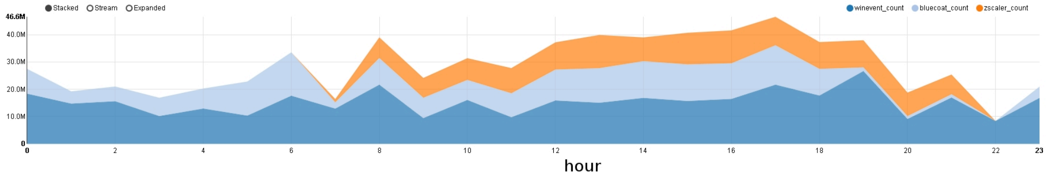


Figure 1: Proxy server logs and Windows Events used for the security analytics

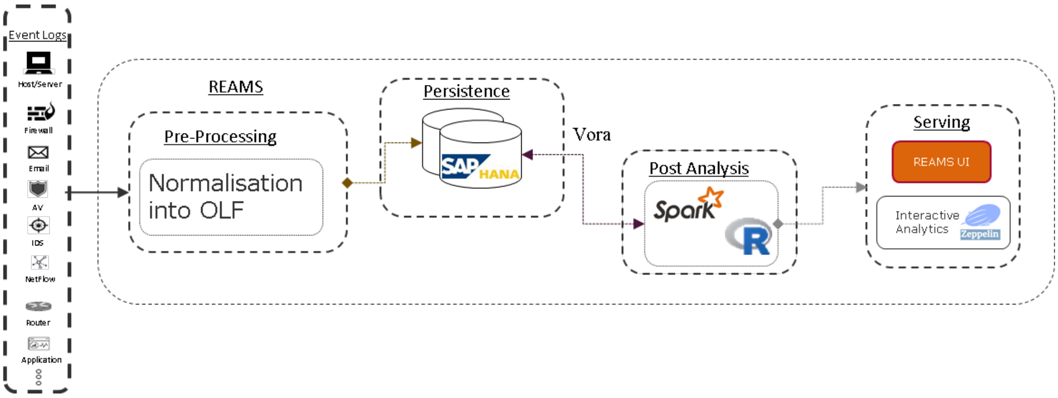


Figure 2: Architecture of Real-time Event Analytics and Monitoring System

The dataset contains 3 data feeds, marked with different colours in the Figure 1. First one contains 368,185,755 Windows Events. Second and third feeds are log messages from proxy servers (Blue Coat [3] and Zscaler [11] with 192,967,831 and 133,816,222 log messages correspondingly). All 3 feeds should cover a period of 24 hours. Unfortunately, the dataset we got has some inconsistencies, for example, part of Zscaler log messages for the first 5 hours is not available.

However, we still were able to analyse all data available to us and describe our approach and results in the sections below.

2 Analytical Environment

To analyse the data, we have utilised a prototype of the SIEM system jointly developed in HPI, namely Real-time Event Analytics and Monitoring System (REAMS) [10]. REAMS is capable of high-speed data aggregation, normalisation and analysis. The architecture of our system is presented in Figure 2.

REAMS supports gathering data from multiple systems including Windows- and Linux-based computers with its own client. Besides this, the REAMS can also collect the data from existing Log Management or SIEM server of the enterprise, as well as supports direct data import from various types of archive. The gathered data are normalised into Object Log Format [9] and stored into SAP HANA database. Thanks to the usage of SAP HANA in-memory database [5], all data are stored and processed directly in the main memory of the database server. The recent versions of SAP HANA support connection with Apache Spark [1] through SAP Vora libraries [8], which allows us to extend the analytical capabilities with algorithms provided through Spark. In addition to the analytical capabilities of SAP HANA and Apache Spark, we also implement our own outlier detection and other analytical methods using SAP HANA R integration [6]. The results of the data analysis are available in the REAMS Graphical User Interface or, alternatively, Apache Zeppelin (an interactive data analytics solution [2]).

To analyse the dataset with our system, we have used two approaches described in the subsections below.

2.1 Correlation of Windows Events and proxy server log messages

We have started our analysis with Windows Event subset of the data and queried the data using SQLSCRIPT language available from SAP HANA database [7]. The simple SQL queries allow us to gather statistics about Windows Events, such as users with failed logon events, account lockouts, unusual activity time, file share accesses and so on. However, such information alone does not indicate the malicious activity, since most of these events could be a result of automated activity (e.g., Windows network discovery) or misconfiguration. Since proxy server logs were also available to us, we decided to correlate them together to find out the users with threat indicators within all data feeds. To achieve it, we have followed the steps below:

- select top most suspicious users, e.g., with the high number of failed logon events and account lockouts from the Windows Event data feed
- for each user, identify his IP address at every particular period of time (since IP addresses in the dataset are changed regularly due to DHCP settings)
- from the proxy server logs, select connections to the malicious domains, listed in public blocklists
- check if there were any connections from IP addresses of the suspicious users to blacklisted domains.

2.2 Hybrid Outlier Detection

Besides correlation algorithm described in the subsection above, we also apply our hybrid outlier detection algorithm, described earlier in [10]. Under this algorithm, all data are first converted into vector space, where each unique value of each column becomes a new column. Next, we randomly select multiple training subsets from this vector space and perform spherical k-means clustering to find clusters in the training data. The concept vectors of these clusters are learned by one-class SVM as normal (we use one SVM model per training subset). Finally, we divide all data into the subsets of the same size, cluster them and check concept vectors of clusters against trained SVM models. The clusters that are classified as outlier by all SVM models are reported and ranked by normalised sum of decision values.

3 Detected anomalies

The correlation algorithm described in subsection 2.1 returned empty result sets, which means that there is no relation between top users with high number of failed logon events and connections to the blacklisted domains. The same holds for the top users with high numbers of lockouts.

Nevertheless, the applied outlier detection returned clusters with anomalies. We have checked several top ranked clusters manually, since we have an access to the SIEM system of the partner enterprise. Among the false positive alerts, we were able to identify the following issues:

- Test user who was only active 3 days and produced logon failure events only
- A user that put his password into username field
- Helpdesk user having 7 logon failures on 2 IP addresses (probably mistyped the password or tried to login on the machine, where he has no right to do it)
- Service account having failed logon events on 1 IP address. We have checked the whole time range available for us in Splunk system and can conclude, that it was the single spike with failed logon events on the date we have analysed. This issue was reported to the security department of the partner enterprise.
- A user with dedicated role having only authentication failures.
- Server accounts locked out or producing failed logon events

Although the listed cases do not represent really malicious activity[1], they prove that the offered outlier detection approach is able to automatically identify suspicious events.

4 Future Work

So far under this project we were able to implement correlation of security-related events between multiple data sources, such as Windows Events and proxy logs. We have also applied our Hybrid Outlier Detection on the heterogeneous Windows Audit Events and proved that our approach produces feasible results.

In the next project phase we plan to extend our correlation algorithms and also apply multiple outlier detection methods on the combined dataset, containing data from all available sources. Besides data sources described in this report, we also plan to include log messages from DHCP server and information from the threat intelligence platform.

[1] We assume that there were no security incidents during the analysed period of time.

5 Conclusion

Thus, under this Future SOC Lab project we have implemented and applied the algorithm for correlation of Windows Events and proxy server logs, as well as applied an existing outlier detection method. The results show no relation between authentication alerts (logon failures and user lockouts) and connections to malicious domains. Nevertheless, the applied outlier detection algorithm allowed us to find suspicious activity in the network, which was manually checked and proved through the SIEM system of the partner enterprise.

References

[1] *Apache Spark - Lightning-fast cluster computing*. URL: https://spark.apache.org/ (last accessed 2017-01-01).

[2] *Apache Zeppelin. A web-based notebook that enables interactive data analytics*. URL: https://zeppelin.apache.org/ (last accessed 2017-01-01).

[3] *Blue Coat Reverse Proxy*. URL: https://www.bluecoat.com/products-and-solutions/reverse-proxy (last accessed 2017-01-01).

[4] K. M. Kavanagh, O. Rochford, and T. Bussa. *Magic Quadrant for Security Information and Event Management*. Technical report G00290113. Gartner, Aug. 10, 2016.

[5] *SAP HANA*. URL: http://www.saphana.com (last accessed 2017-01-01).

[6] *SAP HANA R Integration Guide*. URL: https://help.sap.com/hana/SAP_HANA_R_Integration_Guide_en.pdf (last accessed 2017-01-01).

[7] *SAP HANA SQLScript Reference*. URL: https://help.sap.com/viewer/de2486ee947e43e684d39702027f8a94/2.0.00/en-US (last accessed 2017-01-01).

[8] *SAP Vora*. URL: https://www.sap.com/product/data-mgmt/hana-vora-hadoop.html (last accessed 2017-01-01).

[9] A. Sapegin, D. Jaeger, A. Azodi, M. Gawron, F. Cheng, and C. Meinel. "Hierarchical object log format for normalisation of security events". In: *2013 9th International Conference on Information Assurance and Security (IAS)*. IAS '13. IEEE, Dec. 2013, pages 25–30. ISBN: 978-1-4799-2990-0. DOI: 10.1109/ISIAS.2013.6947748.

[10] A. Sapegin, D. Jaeger, F. Cheng, and C. Meinel. "Towards a system for complex analysis of security events in large-scale networks". In: *Computers & Security* 67 (2017), pages 16–34.

[11] *Zscaler Internet Access*. URL: https://www.zscaler.com/products/zscaler-internet-access (last accessed 2017-01-01).

The Structure of Industrial SAT Instances
Comparing SLS and Backtracking Solvers

Tobias Friedrich, Ralf Rothenberger, and Andrew M. Sutton

Hasso Plattner Institute, Potsdam, Germany
{firstname.lastname}@hpi.de

We continue our ongoing project examining non-uniform random distributions of propositional satisfiability formulas. In this phase of the project, we compare the results of stochastic local search (SLS) and backtracking SAT solvers near the phase transition of scale-free propositional satisfiability instances.

1 Introduction

Propositional satisfiability (SAT) is one of the most fundamental problems in computer science. Many practical questions from different domains can be encoded as propositional formula and solved by determining the satisfiability of the resulting formula. A propositional formula is constructed from a set V of n Boolean variables by forming a conjunction

$$F = C_1 \wedge C_2 \wedge \cdots \wedge C_m$$

of m disjunctive clauses where

$$C_i = (\ell_1 \vee \ell_2 \vee \cdots \vee \ell_{k_i})$$

where $\ell_j \in \{v, \neg v\}$ for some $v \in V$. Here $\neg v$ denotes the logical negation of v. The goal of the decision problem is to decide if there is an assignment to all variables of V so that F evaluates to true. SAT is a central problem in theoretical computer science, but it is also an important practical problem since many difficult combinatorial problems reduce to it.

SAT instances and distributions. A distribution of SAT instances is typically parameterized by n and m and is described by a categorical distribution over all formulas over n variables and m clauses. The most heavily studied distribution of SAT instances is the *uniform* distribution. The uniform distribution is the distribution $U_{n,m}$ of all well-formed CNF formulas on n variables and m clauses where each formula has the same probability of being selected.

Most theoretical work on SAT instances has focused almost exclusively on this uniform distribution $U_{n,m}$. Uniform random formulas are easy to construct, and have shown to be accessible to probabilistic analysis due to their statistical uniformity. Indeed, a long line of successful research has relied on the uniform distribution, and from it, several sophisticated rigorous and non-rigorous techniques have developed for analyzing random structures in general.

Nevertheless, a focus on uniform random instances comes with a risk of driving SAT research in the wrong direction [9] because such instances do not possess the same structural properties as ones encountered in practice. It is well-known that solvers that have been tuned to perform well on one class of instances do not necessarily perform well on another [4], and studying the algorithmics of solvers on uniform random formulas can lead research astray.

The empirical SAT community has expanded their view to study *industrial* instances. Industrial instances arise from problems in practice, such as hardware and software verification, automated planning and scheduling, and circuit design. Empirically, industrial instances appear to have strongly different properties than formulas generated uniformly at random, and as might be expected, SAT solvers behave very differently when applied to them [7, 10].

Furthermore, a number of *non-uniform* random distributions have been recently proposed. These models include regular random [5], geometric [6] and scale-free [1, 2]. The scale-free model is especially promising because the *degree distribution* (distribution of variable occurrence) of instances follows a power-law and this phenomenon has been observed on real-world industrial instances.

Project aim. The goal of this phase of the project was to utilize the parallel computing power of the 1000 node cluster of the Future SOC Lab to (1) generate a massive set of large random non-uniform (scale-free) formulas and check their satisfiability & hardness and (2) compare backtracking and SLS solvers in their ability to determine bounds on the satisfiability threshold for different values of power-law exponent.

2 Empirical scale-free thresholds

In contrast to the uniform random model, in which each formula on n variables and m clauses is drawn uniformly at random from all such formulas, the scale-free random model generates formulas so that the fraction of variables occurring z times is proportional to $z^{-\beta}$, where β is a parameter known as the *power-law exponent*.

The power-law exponent adds a further degree of freedom to the distribution of formulas, and we want to understand the satisfiability threshold corresponding to the distribution. This is the point, as a function of constraint density, measured by m/n, or the number of clauses divided by the number of variables, where the probability that a formula is satisfiable drops from 1 to 0.

Table 1: Empirically determined satisfiability threshold values for each power-law value.

β	threshold
3.0	3.3375
3.5	3.76
4.0	3.927 27

We begin by trying to empirically determine the satisfiability threshold for chosen values for β. In particular, for $\beta = 3, 3.5, 4$, we generate a number of formulas at varying densities and execute a SAT solver (`MapleCOMSPS`) to determine whether it is satisfiable. We take n from 100 to 1000 in steps of 50, however, as β increases, formulas near the

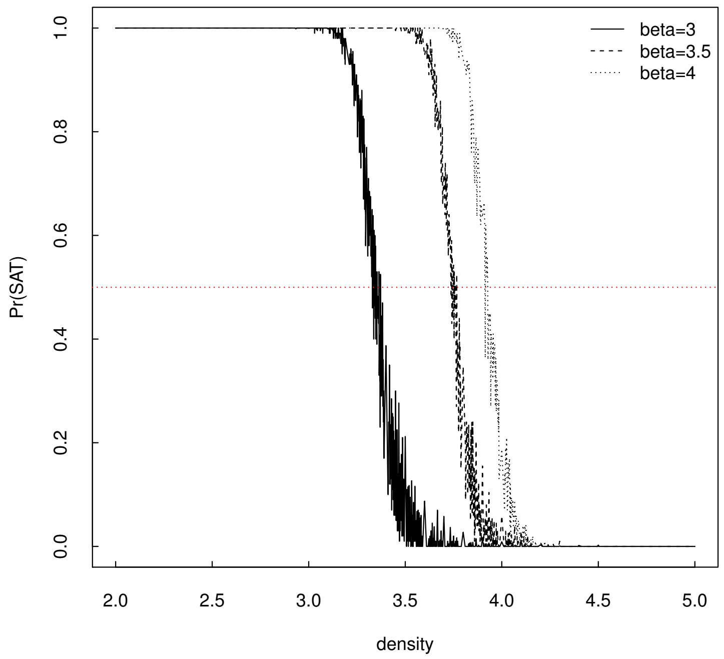


Figure 1: Determining the empirical satisfiability threshold for three values of power-law exponent as a function of clause density. The value of n varies from 100 to 1000.

threshold become harder to solve, and thus we are only able to produce formulas up to $n = 850$ for $\beta = 3.5$ and $n = 700$ for $\beta = 4$.

For each n, a sequence of clause lengths is computed, and for each clause length, 200 formulas were generated and checked for satisfiability with MapleCOMSPS. To determine the threshold, we take the density value that yields the proportion of satisfiable formulas that is closest to $1/2$. The result of these experiments is plotted in Figure 1. The empirically determined values are listed in Table 1.

3 SLS versus a backtracking solver

We are interested in the influence of power-law exponent on different styles of SAT solver at the threshold. We measured the performance of the following solvers.

1. MapleCOMSPS [11]: a CDCL backtracking solver based on MiniSAT [8] that implements machine learning in its branching heuristics. Both MapleCOMSPS and MiniSAT have performed well on industrial benchmarks.

2. WalkSAT [12]: a simple stochastic local search (SLS) solver that is based on a conflict-directed random walk.

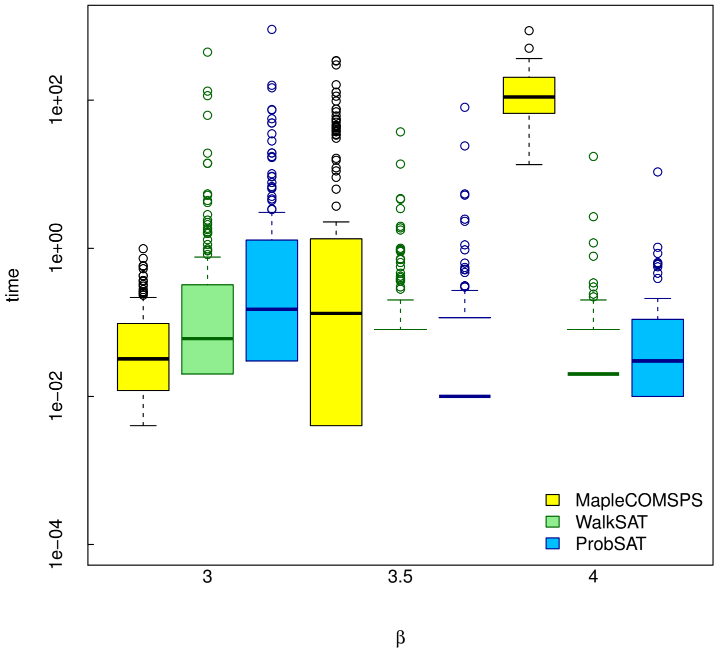


Figure 2: Comparison of SLS and backtracking solvers at the satisfiability threshold for different β values for satisfiable power-law formulas. Note that the y-axis is log-scaled. Thus, we see a heavy dependence of runtime on β for the backtracking solver. The SLS solvers have a less pronounced, but negative trend.

3. `probSAT` [3]: a simple probabilistic SLS solver that computes a distribution over variables to flip based on make and break counts.

We compared the above solvers applied to formulas generated at the thresholds determined empirically. We used GNU Parallel [13] to distribute a large number of jobs over the cluster. Each job was responsible for generating a set of random scale-free formulas, and then attempting to solve each with each of the solvers listed above within a predetermined time limit. The solvers `WalkSAT` and `probSAT` only work on satisfiable instances, so we first needed to remove unsatisfiable instances by checking the output of the backtracking solver (`MapleCOMSPS`). The timings for the unsatisfiable instances are removed from the output of the backtracking solver. A grouped boxplot of the solvers is shown in Figure 2.

4 Conclusions

We were able to compare the behavior of a number of different SAT solvers along the phase transition of the power law distribution by executing a massive number of solvers in parallel across the FSOC cluster. We found a strong runtime dependence on

β at the threshold for backtracking solvers (larger β corresponds to longer runtimes), whereas the SLS solvers seemed to have a weaker negative trend with β.

References

[1] C. Ansótegui, M. L. Bonet, and J. Levy. "Towards Industrial-like Random SAT Instances". In: *Proceedings of the 21st International Jont Conference on Artifical Intelligence*. IJCAI'09. Pasadena, California, USA: Morgan Kaufmann Publishers Inc., 2009, pages 387–392.

[2] C. Ansótegui, M. L. Bonet, and J. Levy. "On the Structure of Industrial SAT Instances". In: *Principles and Practice of Constraint Programming - CP 2009*. Edited by I. P. Gent. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pages 127–141. ISBN: 978-3-642-04244-7. DOI: 10.1007/978-3-642-04244-7_13.

[3] A. Balint and U. Schöning. "probSAT and pprobSAT". In: *SAT Competition* 2014 (2014), page 63.

[4] M. Birattari. *Tuning Metaheuristics*. Springer Berlin Heidelberg, 2009. ISBN: 978-3-642-00482-7. DOI: 10.1007/978-3-642-00483-4.

[5] Y. Boufkhad, O. Dubois, Y. Interian, and B. Selman. "Regular random k-SAT: properties of balanced formulas". In: *J. Automat. Reason.* 35.1-3 (2005), pages 181–200. ISSN: 0168-7433. DOI: 10.1007/s10817-005-9012-z.

[6] M. Bradonjić and W. Perkins. "On sharp thresholds in random geometric graphs". In: *Approximation, randomization, and combinatorial optimization*. Volume 28. LIPIcs. Leibniz Int. Proc. Inform. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2014, pages 500–514.

[7] J. P. Crawford. "Local automorphism invariance: gauge boson mass without a Higgs particle". In: *J. Math. Phys.* 35.6 (1994), pages 2701–2718. ISSN: 0022-2488. DOI: 10.1063/1.530532.

[8] N. Eén and N. Sörensson. "An Extensible SAT-solver". In: *Theory and Applications of Satisfiability Testing*. Edited by E. Giunchiglia and A. Tacchella. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pages 502–518. ISBN: 978-3-540-24605-3. DOI: 10.1007/978-3-540-24605-3_37.

[9] H. Kautz and B. Selman. "The state of SAT [Fourth International Symposium on the Theory and Applications of Satisfiability Testing]". In: *Discrete Applied Mathematics. The Journal of Combinatorial Algorithms, Informatics and Computational Sciences* 155.12 (2007). Held at Boston University, Boston, MA, June 14–15, 2001, pages 1514–1524. ISSN: 0166-218X. DOI: 10.1016/j.dam.2006.10.004.

[10] K. Konolige. "Easy to be Hard: Difficult Problems for Greedy Algorithms". In: *Principles of Knowledge Representation and Reasoning*. Edited by J. Doyle, E. Sandewall, and P. Torasso. The Morgan Kaufmann Series in Representation and Reasoning. Morgan Kaufmann, 1994, pages 374–378. DOI: 10.1016/B978-1-4832-1452-8.50130-5.

[11] J. H. Liang, C. Oh, V. Ganesh, K. Czarnecki, and P. Poupart. "MapleCOMSPS, MapleCOMSPS_LRB, MapleCOMSPS_CHB". In: *Proceedings of SAT Competition 2016; Solver and Benchmark Descriptions*. Edited by T. Balyo, M. J. H. Heule, and M. Järvisalo. 2016, page 52. ISBN: 978-951-51-2345-9. HDL: 10138/164630.

[12] B. Selman, H. A. Kautz, and B. Cohen. "Noise Strategies for Improving Local Search". In: *Proceedings of the 12th National Conference on Artificial Intelligence, Seattle, WA, USA, July 31 - August 4, 1994, Volume 1.* Edited by B. Hayes-Roth and R. E. Korf. AAAI Press / The MIT Press, 1994, pages 337–343. ISBN: 0-262-61102-3.

[13] O. Tange. "GNU parallel: The Command-Line Power Tool". In: *;login:* 36.1 (2011), pages 42–47.

Revealing Infrastructure-Stressing Clients in the Customer Base of a Scandinavian Operator using and HPI Future SoC Lab Hardware Resources

Julia Sidorova[1], Lars Lundberg[1], Oliver Rosander[1], and Lars Skold[2]

[1] Blekinge Institute of Technology, Karlskrona, Sweden
julia.a.sidorova@gmail.com
lars.lundberg@bth.se
[2] Telenor AB, Stockholm, Sweden
lars.skold@telenor.se

We define the term an Infrastructure-Stressing client. Roughly speaking, she uses the infrastructure in a taxing manner, such as always staying in the zones of high demand. We developed a method based on combinatorial optimization to reveal the Infrastructure-Stressing clients in the customer base based on trajectory information from Call Data Records. We have found that 7 % in the customer base are Infrastructure-Stressing. As was expected, a correlation exists between this quality and client's geo-demographic segment. Currently we are working on a predictive model to be able to tell an Infrastructure-Stressing client in a newcomer whose mobility is yet unknown to the operator.

1 Introduction

In telecommunication industry, the lion's share of capital is spent on the infrastructure and its maintenance. The revenues are dependent on the size and quality of the customer base: without a customer there is no business, yet satisfying all customers is simply not feasible, unless the right ones are chosen and others are let go. While designing the principles for identifying bad customers, geographical realities in the region, and infrastructure need to be taken into account.

In business analytics industries rely heavily on commercial geo-demographic segmentation systems (MOSAIC, ACORN, etc.), which are a universally strong predictor of user's behavior: from diabetes propensity and purchasing habits to political preferences. A segment is assigned based on a postcode of the client's home address. The client's home address is a surprisingly powerful predictor of client's behavior. People of similar social status and lifestyle tend to live close. Compared with conventional occupational measures of social class, postcode classifications typically achieve higher levels of discrimination, whether averaged across a random basket of behaviors recorded on the Target Group Index or surveys of citizen satisfaction with the provision of local authority services. One of the reasons that segmentation systems like MOSAIC are so effective is that they are created by combining statistical averages for both census data and consumer spending data in pre-defined geographical units [2]. The postcode descriptors allow us powerful means to unravel lifestyle differences in ways that are difficult to distinguish using conventional survey research given limited sources and sample size constraints [10]. For example, it was demonstrated that middle-class MOSAIC categories in the UK such as 'New Urban Colonists', 'Bungalow Retirement', 'Gentrified Villages' and 'Conservative Values', whilst very similar in terms of overall social status, nonetheless register widely different public attitudes and voting intentions, show support for different kinds of charities and preferences for different media as well as different forms of consumption. Geodemographic categories correlate to diabetes propensity [4], school students' performance [10], broadband access and availability [2] and so on. Industries rely increasingly on geodemographic segmentation to classify their markets when acquiring new customers [3]. The localized versions of MOSAIC have been developed for a number of countries, including the USA and the EU countries. The main geodemographic systems are in competition with each other and the exact details of the data and methods for generating lifestyles segments are never released [1] and, as a result, the specific variables or the derivations of these variables are unknown.

Apart from place of residence segmentation, there is an individual aspect to every client and her characteristic footprint on the infrastructure, which is a result of interplay of many factors: individual mobility patterns, concrete infrastructure of the telecommunication operator and relative mobility of other users. In this work, a conclusion via combinatorial optimization is made about the client's impact on the infrastructure exploitation.

The rest of the report is structured as follows: Section 2 describes the data set. Section 3 describes the proposed method. Section 4 covers experimental findings, and finally conclusions are drawn in Section 5.

2 Data Set

The study has been conducted on anonymized geospatial and geo-demographic data provided by a Scandinavian telecommunication operator. The data consists of CDRs containing historical location data and calls made during one week in 2015 in a midsized city in Sweden with more than thousand radio cells. Several cells can be located on the same antenna. Their density varies in different areas and is higher in city centers, compared to rural areas. The location of 27010 clients is registered together with which cell serves the client. The location is registered every five minutes. In the periods when the client does not generate any traffic, she does not make any impact on the infrastructure and such periods of inactivity are not included in the analysis. We reconstructed a user's trajectory based on the time-ordered list of cell phone towers from which a user made her calls or send an SMS. Every client in the database is labeled with her MOSAIC segment. There are 13 segments present in the region. The fields of the database used in this study are:

- the cells IDs from which a user made her calls,
- the location coordinates of cells,

- the time stamps of every event, and
- MOSAIC geo-demographic segment for each client.

3 A Method to Reveal Infrastructure-Stressing Clients

There are Infrastructure-Stressing and friendly clients, where the former use infrastructure in a taxing manner, such as always staying in the zones of high demand and the latter predominantly stay in the zones of low demand, where the cells are idle. The algorithm to reveal the Infrastructure-Stressing clients is below. The method uses a combinatorial optimization function, which returns the coefficients reflecting the desirability of the client segment, where the 0 value corresponds to an *absolutely unwanted* interpretation. Appendix 1 details the combinatorial optimization function.

```
Input: Data with user mobility: <$User_{ID,}$ time stamp t, cell j that serves the client>.
[I. With a numeric value, characterize the users with respect to their mobility areas.]

1. For each user $User_{ID}$, an array $Counts_{ID}$ with 2016 elements (7 days × 24 hours × 12 five-minute slots) is generated:
      $Counts_{ID}$: $N_{1,}$ $N_2$,…,$N_{2016}$.
   The elements $N_t$ in the array are the counts of the number of users that are being served by the same cell j at the same time t.
2. For each user $User_{ID}$, the elements in the array $Counts_{ID}$ are sorted in a decreasing order.
      For each ID{
         Array $sortedCounts_{ID}$=sort($Counts_{ID}$)
      }
3. For every client, sum up the top 5% of elements from the array $sortedCounts_{ID}$ (100 elements).
      For each ID{
         $Value_{ID}=\Sigma_{k=1..100}N_{ID,k}$;
         HashMap $value_{ID}$=<$User_{ID}$, $value_{ID}$>
      }
4. Sort the data structure $Value_{ID}$ by the field of the $value_{ID}$
      For every ID{
         $sortedValue_{ID}$=sort $_{by\ ValueID}$ < $User_{ID}$,$value_{ID}$>
      }

[II. Initialization Steps]
5. Set $x_{stressful}$ to 0.
6. Set the array A containing the IDs of stressing clients to ∅.
      Array A = ∅.
[III. Reveal the Infrastructure-Stressing clients.]
7. While ($x_{stressful}$ = 0) do {
   [Tentatively label the users into Infrastructure-Stressing, friendly and medium]
```

```
 8. The top 1% of clients (100 clients) are labeled into the Infrastructure-
    Stressing segment
        $ID_{stressing?}$ = top 1% of the user list.
 9. The bottom 1% of the users are labeled into the infrastructure-friendly
    segment.
        $ID_{friendly?}$ = bottom 1% of the user list.
10. Other users are assigned to the medium segment.
[Check the tentative division with via the combinatorial optimization module]
11. The combinatorial optimization model is solved for the three segments
    representing different attractiveness classes to get the optimal coeffi-
    cients for their proportion and obtain $\mathbf{x}$ = { $x_{stressing}$, $x_{medium}$, $x_{friendly}$
    }.
        ($x_I$,$max\_obj_{I,D}$) = combinatorial_optimization(I,D)
12. IF ($x_{stressing}$ = 0), THEN add the IDs of the clients from the stressful
    segment into the array with Infrastructure-Stressing clients A.
        $ID_{stressing}$=$ID_{stressing}$ + $ID_{stressing?}$
13. Remove the records with the stressing clients from the database.
}[end of while]
Output: the list with the IDs of the Infrastructure-Stressing clients.
```

4 Experiments

In the first round of the *while* loop, the extremes of Infrastructure-Stressing and Infrastructure-Friendly clients were processed, and the vector with normalized scaling coefficients was returned to be
$[x_{stressing}, x_{medium}, x_{friendly}]=[0, 0.3, 0.7]$.

In line with expectations, the value of the scaling coefficient equal to *0* suggests that the extremely Infrastructure-Stressing clients are absolutely unwanted [7, 8], i.e.
$f_{wanted}(extremely\_stressing) = 0.$

The list with UserIDs of Infrastructure-Stressing clients has been obtained. Seven percent of the customer base was revealed to be Infrastructure-Stressing.

As was expected, a correlation exists between this quality and client's geo-demographic segment. There are 13 MOSAIC groups present in the database. Six of them are less than 0.05 probable to have Infrastructure-Stressing client, while the maximum probability for a segment to contain an Infrastructure-Stressing client is 0.2.

5 Conclusions and Next Steps

We have proposed the notion of an *Infrastructure-Stressing Client* and revealed a list of those in the Telenor Sweden customer base based on historical user mobility and the individual infrastructure characteristics. We have found that 7 % in the customer base are Infrastructure-Stressing. As was expected, a correlation exists between this quality and client's geo-demographic segment. Currently we are working on a predictive model to be able to tell an Infrastructure-Stressing client in a newcomer, whose mobility is yet unknown to the operator.

A Appendix 1

As input the combinatorial optimization module takes the matrix **A** of size $N \times M$, where N is the number cell towers × number of time slots (there are 2016 five minute in one week), and M number of geodemographic segments. Every cell $\mathbf{A}_{\mathrm{i,j}}$ contains the footprint by the users from geodemographic segment *j*. As output the recommendation is given about the optimal proportion of geodemographic segments in the customer base. An LP formulation is given to the problem is as follows:

- *the objective function* maximises the number of clients,
- *the decision variable*s are the scaling coefficients to boost or to reduce the number of clients in each of the segments, and
- *the restrictions* are formulated not to overload any cell at any time, i.e. that the historical footprint of the segment scaled with the coefficent is less than the cell's capacity.

Let us define the model's variables:

- *Ik: the set with geo-demographic segments {segment1, ..., segmentk};*
- *D:* the mobility data for a region that for each user contain client's ID, client's geo-demographic segment, time stamps when the client generated traffic, and which antenna served the client at a particular time stamp.
- S_i: the number of subscribers that belong to a geo-demographic segment *i*;
- $S_{i,t,j}$: the number of subscribers that belong to a geo-demographic segment *i*, who are using the network at some time moment *t*, being registered with a particular cell *j*;
- C_j: the capacity of cell *j* in terms of how many persons it can safely handle simultaneously;
- $\boldsymbol{x}$: the vector with the scaling coefficients for the geo-demographic segments.
- *The vector* $\boldsymbol{x}$ *with the decision variables*

$$\boldsymbol{x}=\{\mathrm{x}_{\mathrm{segment1}}, .., \mathrm{x}_{\mathrm{segmentM}}\}.$$

The decision variables represent the scaling coefficients for each geo-demographic segment. For example, for the category in the clientele that is to be doubled $x_i = 2$. Similarly, if $x_i < 1$ for a geo-demographic segment, it means that the number of clients is to be reduced.

- *The objective function* seeks to maximize the number of subscribers:

$$Maximize\ \Sigma_{i=1..15}\ S_i\ x_{i.}$$

- *The restrictions*

$$for\ all\ t,j,\ \Sigma_{i=1..15}\ S_{i,t,j}\ x_i \leq C_j.$$

A consensus reached in the literature [5, 6, 9] is that the mobility pattern for the subscribers is predictable due to strong spatio-temporal regularity. The corollary is that the increase in the number of subscribers in a given segment with a factor x will result in an increase of the load generated by the segment with a factor x for each time and cell.

The value of the objective function is sensitive to the number of segments: the more segments, the more degrees of freedom the LP has, and the higher the objective value is. There are 13 geo-demographic MOSAIC segments present in the database. A good decision support discovers and prompts actions that lead to revenues, and different actions suggested are comparable profit-wise: among competing modules the one that reveals more profitable actions is better. Even when the optimal proportion of segments in the customer base can be unreachable for various reasons, the optimal proportions indicate a direction of the maximum revenue and most efficient infrastructure exploitation.

References

[1] J. Debenham, G. Clarke, and J. Stillwell. "Extending geodemographic classification: a new regional prototype". In: *Environment and Planning A* 35.6 (2003), pages 1025–1050.

[2] T. H. Grubesic. "The geodemographic correlates of broadband access and availability in the United States". In: *Telematics and Informatics* 21.4 (2004), pages 335–358.

[3] M. Haenlein and A. M. Kaplan. "Unprofitable customers and their management". In: *Business Horizons* 52.1 (2009), pages 89–97. ISSN: 0007-6813. DOI: 10.1016/j.bushor.2008.09.001.

[4] J. Levy. *How to Market Better Health – Diabetes. A Dr. Foster Community Health Workbook*. London: Dr. Foster Intelligence, 2004.

[5] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson. "Approaching the limit of predictability in human mobility". In: *Scientific reports* 3 (2013), page 2923.

[6] D. Naboulsi, M. Fiore, S. Ribot, and R. Stanica. “Large-scale mobile traffic analysis: a survey”. In: *IEEE Communications Surveys & Tutorials* 18.1 (2016), pages 124–161.

[7] C. Niyizamwiyitira, L. Skold, L. Lundberg, and J. Sidorova. *Analytic queries on Telenor data*. HPI FutureSOC Lab Day. 2016.

[8] J. Sidorova, L. Lundberg, and L. Skold. “Optimizing the Utilization in Cellular Networks using Telenor Mobility Data and HPI Future SOC Lab Hardware Resources”. In: *HPI Future SOC Lab: Proceedings 2016*. Edited by C. Meinel, A. Polze, G. Oswald, R. Strotmann, U. Seibold, and B. Schulzki. Potsdam, Germany, 2016.

[9] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. “Limits of predictability in human mobility”. In: *Science* 327.5968 (2010), pages 1018–1021.

[10] R. Webber and T. Butler. “Classifying pupils by where they live: how well does this predict variations in their GCSE results?” In: *Urban Studies* 44.7 (2007), pages 1229–1253.

Computational Fluid Modeling to setup the thermal control of an astronomical instrument for polarimetry detection (GRANTECAN polarimeter)

Igor Di Varano

Leibniz Institut für Astrophysik, Potsdam
idivarano@aip.de

The target of the current project is that of setting the proper environmental conditions surrounding an astronomical instrument, in the particular case GRAPE (a polarimeter for the main Cassegrain focus of Grantecan telescope). The modern approach in designing astronomical instrumentation, in the context of integrated modeling approach [5], requires since the beginning the definition of the thermal architecture: in facts the astronomers are always interested in how the seeing induced by the telescope structure via the convective layer and cold sky radiation affects the optical performances of the instrument. Hereafter we present the strategy we intend to use.

1 Introduction

GRAPE (Grantecan Polarimeter) is a polarimeter planned to be installed on the main Cassegrain focus of GTC (Gran Telescopio Canarias), but also, depending on their availability, with an open possibility to go to one of the folded Cassegrain foci.

The telescope, which is located at the Observatorio del Roque de los Muchachos (ORM), in La Palma, Canary Islands, at an altitude of 2267 meters on sea level, has an equivalent entrance pupil of 10.4 m, with a primary mirror consisting of 36 regular hexagonal segments, each with a side length of 936 mm; it has a focal distance of 16.500 mm, magnification factor of 10.3 and back focal distance equivalent to 3400 mm [2]. The instrument will provide full Stokes polarimetry, feeding HORS (High Optical Resolution Spectrograph) via a fiber link of Ø300 µm. The spectrograph, which has been recently commissioned, is located on the Nasmyth platform, with FWHM resolving power of about 25,000 (5 pixel) and a spectral range of 400 nm to 680 nm.

2 Motivation

Evidently for every instrument we want to design, we wish to keep it thermally stable within a certain tolerance, which depends on the wavelength coverage of the detector we are going to use (visible or IR), in its turn selected depending on the science cases. For instance to detect radial velocities of the order of 1 /ms we definitely need a vacuum system. In the case of our polarimeter the requirements are rather more relaxed, as those of a photometer, meaning that it' s necessary to keep the temperature and pressure fluctuation to a tenth of degree and a tenth of mbar. The main issue for the CFD analysis has revealed to be the complex topology, more than the boundary conditions assignment. Namely it was possible to proceed adopting a few simplifications. Two separate fluid domains have been identified, one corresponding to the environment, outside the observing instrument, the other internal going from the entrance diaphragm at the Cassegrain focal plane down to the fiber output. We have currently completed the preprocessing phase consisting in the definition of the outer volume and identified the inlet, outlet and wall surfaces, plus the internal surfaces where the boundary conditions are applied. We found the meshing process rather demanding and it is not yet optimized. The planned strategy is better illustrated in the diagram of Figure 1.

3 Setting up the model

Two main packages have been identified for CAD processing: FreeCAD and SALOME platform. They both present the advantage to accept Python scripts so that many parameters can be quickly run and automatically updated. I have been working on the original 3Dmodel provided in STEP format by the IAC (Astrophysical Institute of Canary Islands), namely acknowledging J. Rasilla and L. Cavaler. The original model contains all solid bodies, i.e. not shells nor unidimensional beams.

3.1 Topological model

We have decided to use SALOME platform, in particular the *Geometry* package for topology modification and meshing tools. The topology has been edited and repaired since there were many solid intersections, irregular shapes and hollow bodies with open contours which may have led to a failure in generating the embedding fluid region. In particular for preliminary repair the *RemoveInternalFaces* and *RemoveExtraEdges* have been used. With the powerful *Explode* function it has been possible to split the assembly into the components (860 solid bodies). Then by applying several times such Boolean operations as *Fuse* and *Cut* we have been able to get a meaningful assembly to work with. Some bodies of less relevance have been removed: the lateral mechanical structure of the Nasmyth platforms has been kept, together with some components of the triangular whiffle tree structure holding the segments of the primary mirror, the electronic and auxiliary devices and, more relevant, the central supporting trusses and flange of the main Cassegrain focus which hosts our polarimeter. The external fluid embedding the telescope structure and the outer frame of the polarimeter together with the interface flange, electronic and auxiliary boxes has been implemented as an extruded cylinder of 30 m diameter [2], 14 m high. Four inlet openings 4 m × 4 m size corresponding to the 4 cardinal points and one outflow on the top surface of the cylinder Ø10.4 m have been defined. A sequence of multiple *Cut* operations between the cylinder and the fused parts have been executed with final removal of the resulting fluid volumes inside the hollow parts that can be neglected in the calculation. Since all the bodies are included in the cylinder this operation is equivalent to the determination of their complementary. If we consider all the bodies now distinct, after having removed their reciprocal intersection, and naming a single body a, B the enclosure, we get: if $a_i \in A,\ where\ A \subseteq B,\ \therefore \bigcup \overline{a_i} = \bigcup (B \setminus a_i)$.

Then the fluid region has been split in 4 partitions (see Figure 2) to provide an appropriate symmetric meshing and also for a better control on the computational procedure, with the opportunity, increasing the accuracy, to run the process in parallel on the four virtual machines.

3.2 Meshing procedure

In SALOME there are different meshing tools provided, with top-down, bottom-up approach which are even different from one release to the other. Both methods *Gmsh 3D* and *Netgen 3D* parameters have been explored and because of the more reliable and ample documentation, we have finally selected the second option. In general the best recommended way to mesh a fluid is achieved via an automatic tetrahedralization with viscous layers applied on the walls, but since this technique has failed after many approaches and the only successful option has been the plain *Netgen3D*, we kept the last one.

In order to transfer the information of the walls, inlet, outlet they have to be renamed again inside the Mesh environment of SALOME.

The size resulting from the meshing process of a single volume partition is ca. 950 MB, therefore we foresee 3.8 GB for the input data of OpenFoam.

Nevertheless, even if the option to go directly from SALOME into OpenFoam looks like the most comfortable, there are also alternative solutions, which are under investigation, such as Meshlab [4], mostly used for unstructured triangular meshes, so not ideal for fluid domains. It contains many useful topology optimization and reselection filters, like the ones to remove overlapping faces and vertices, smoothing and reduction of existing meshes. Another possibility is given by the *SnappyHexmesh*, which is already integrated within OpenFoam: it exists a more user friendly add-on inside Blender. The ultimate alternative, which is probably the best tradeoff is to use directly OpenFoam for meshing, in particular the *blockmesh* command.

The software developers recommend to provide a regular box containing the geometry we want to mesh, since the meshing tool employs resizable hexahedra.

3.3 Boundary conditions definition

Concerning the input parameters needed to run the simulation, information concerning the available weather station data has been collected, referring to the GTC webpage, which mentions such telescope locations as NOT, TNG and MAGIC.

In particular we have selected for the completeness of dataset the Nordic Optical Telescope (NOT), given that it is possible to save data series specifying such parameters as pressure, humidity, dust, precipitation.

In particular using a Python script, we have stored temperature, pressure, wind velocity and direction, with timestamps every half an hour over a period of one year.

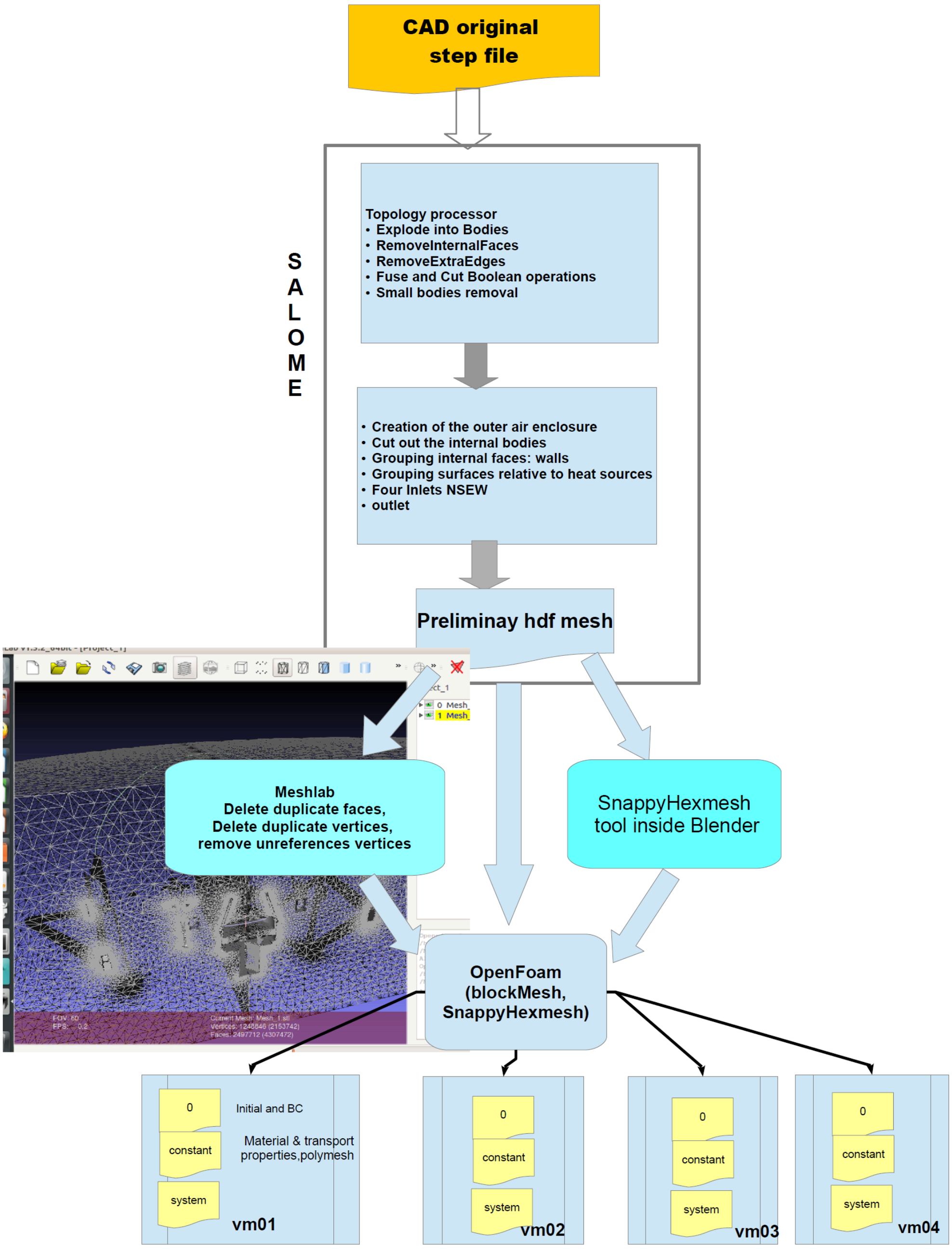


Figure 1: Block diagram of the preprocessor model from the original input down to the delivery of the meshed regions with applied boundary conditions, ready to be inserted into the solver

Then we have selected three nights showcasing the highest temperature peak, the lowest temperature and wind speed peak as the most meaningful to characterize the thermal design of our instrument. In agreement with literature [1] it has been proved that during the night there are two main wind directions. We have chosen the outer faces of 9 electronic cabinets distributed around the focal station, including the 2 belonging to GRAPE, assigning to them a dissipation of 350 $\mathrm{Wm^{-2}}$. The rest of the internal faces have been grouped to apply them a convection coefficient air/steel $h = 8\,\mathrm{Wm^{-2}K^{-1}}$.

The coldest night registered in one year period, starting from April 2016, is the night of 12.02.2017 with $T_{\mathrm{min}} = -6.9\,°\mathrm{C}$, the warmest one was 15.07.2016 with a peak temperature $T_{\mathrm{max}} = 23.9\,°\mathrm{C}$. The maximum wind speed registered in the same period was 32.2 m/s on 23.12.2016.

4 Next steps

We aim to conclude the project passing from the preprocessing phase to the solution and postprocessing steps. We want to assess the temperature and pressure values for the three outlined scenarios, taking into account k-ε turbulence effect, and buoyancy, comparing the various numerical methods in OpenFoam with what is offered by other commercial softwares, as for instance Ansys CFX.

5 Conclusions

Furthermore we would like to continue starting a more ambitious project, based on similar criteria already adopted for GRAPE, entitled "CFD simulations to investigate the thermal conditions and seeing performances of the polarimetric module for E-ELT HIRES (High Resolution Spectrograph for European Extremely Large Telescope)."

Leibniz Institut für Astrophysik is contributing to one of the first generation instrument for E-ELT HIRES with the design of a polarimetric subunit and in the definition of the fiber link between the so called Front End (located on one of the Preliminary Focal Stations on the Nasmyth platform) and the entrance slit to the spectrograph [3]. The project, developed by a consortium of several European countries together with Chile and Brazil, under the leadership of Italian agency INAF, is currently undergoing a two years Phase A study, that will end in March 2018. The telescope and dome design is not finalized yet and a full 3D model will be available after the preliminary design review in Fall 2017. In the meanwhile we're planning to define our own 3D sample, based on a virtual assembly only available in view mode: this will allow us to create straightforward geometries that can be furthermore simplified or suppressed, depending on the encountered meshing issues.

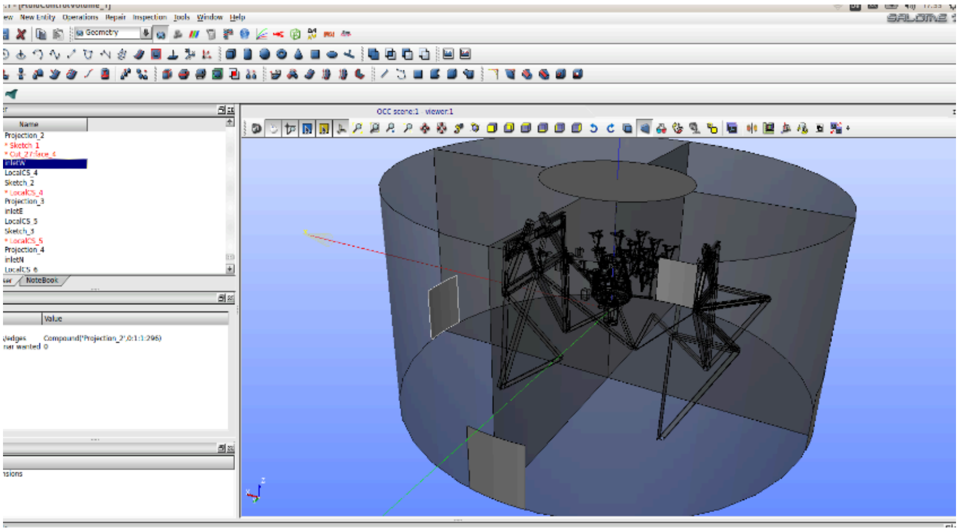


Figure 2: Fluid region with four inlets and one outlet on the top set as boundary conditions for the CFD analysis

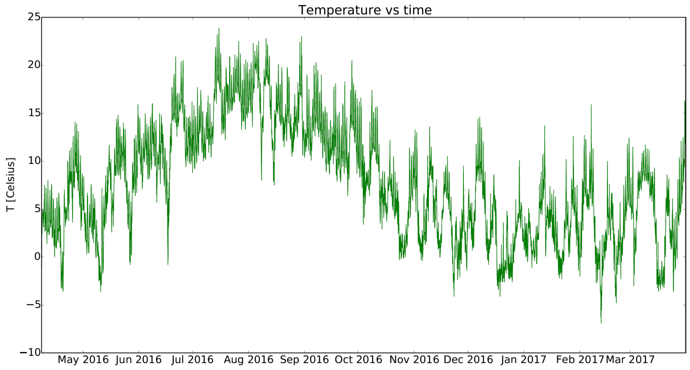


Figure 3: Plot of the temperature recorded at NOT over the period April 2016-17

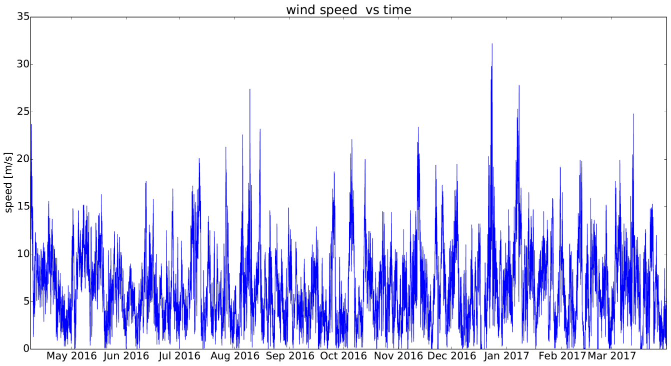


Figure 4: Plot of the wind speed recorded at NOT between April 2016 and April 2017

References

[1] R. Codina, C. Morton, E. Oñate, and O. Soto. “Numerical aerodynamic analysis of large buildings using a finite element model with application to a telescope building”. In: *International Journal of Numerical Methods for Heat & Fluid Flow* 10.6 (2000), pages 616–633. DOI: 10.1108/09615530010347196.

[2] L. Jochum, J. Castro, and N. Devaney. “Gran Telescopio Canarias: current status of its optical design and optomechanical support system”. In: *Advanced Technology Optical/IR Telescopes VI*. Edited by L. M. Stepp. Volume 3352. SPIE, Aug. 25, 1998. DOI: 10.1117/12.319300.

[3] A. Marconi, P. D. Marcantonio, V. D’Odorico, S. Cristiani, R. Maiolino, E. Oliva, L. Origlia, M. Riva, L. Valenziano, F. M. Zerbi, M. Abreu, V. Adibekyan, C. A. Prieto, P. J. Amado, W. Benz, I. Boisse, X. Bonfils, F. Bouchy, L. Buchhave, D. Buscher, A. Cabral, B. L. C. Martins, A. Chiavassa, J. Coelho, L. B. Christensen, E. Delgado-Mena, J. R. de Medeiros, I. D. Varano, P. Figueira, M. Fisher, J. P. U. Fynbo, A. C. H. Glasse, M. Haehnelt, C. Haniff, C. J. Hansen, A. Hatzes, P. Huke, A. J. Korn, I. C. Leão, J. Liske, C. Lovis, P. Maslowski, I. Matute, R. A. McCracken, C. J. A. P. Martins, M. J. P. F. G. Monteiro, S. Morris, T. Morris, H. Nicklas, A. Niedzielski, N. J. Nunes, E. Palle, P. M. Parr-Burman, V. Parro, I. Parry, F. Pepe, N. Piskunov, D. Queloz, A. Quirrenbach, R. R. Lopez, A. Reiners, D. T. Reid, N. Santos, W. Seifert, S. Sousa, H. C. Stempels, K. Strassmeier, X. Sun, S. Udry, L. Vanzi, M. Vestergaard, M. Weber, and E. Zackrisson. “EELT-HIRES the high-resolution spectrograph for the E-ELT”. In: *Ground-based and Airborne Instrumentation for Astronomy VI*. Edited by C. J. Evans, L. Simard, and H. Takami. SPIE, Aug. 2016. DOI: 10.1117/12.2231653.

[4] *Meshlab*. URL: http://meshlab.sourceforge.net (last accessed 2016-01-01).

[5] I. D. Varano, K. G. Strassmeier, M. Woche, and U. Laux. “An integrated thermo-structural model to design a polarimeter for the GTC”. In: *Integrated Modeling of Complex Optomechanical Systems II*. Edited by M. Riva. SPIE, July 2016. DOI: 10.1117/12.2199910.

A Big Data Science Experiment
Protecting Minors on Social Media Platforms

Estée van der Walt and Jan H. P. Eloff

Department of Computer Science
Security & Data Science Research Group
University of Pretoria, South Africa
estee.vanderwalt@gmail.com,eloff@cs.up.ac.za

The detection of identity deception is important for a variety of reasons. One of these being threats to individuals. The research at hand proposes to focus on the protection of minors on big data platforms, like social media, as a use case for identity deception detection. Much work has been done on the detection of fake accounts pertaining to bots. Current research, however, is still lacking in identifying actual people lying about who they are. It necessitates an intelligent identity deception indicator to automate the detection of such deception.

1 Project idea

Identity deception on social media platforms is very important for a variety of reasons. It is gaining increasingly attention. Identity deception occurs when the truth is misrepresented to assume the identity of another. This differs from deception in general in that identity deception is a subset of deception. Examples of other types of deception, like content deception, is when someone is lying about what they did this morning or where they were last night [9].

Identity deception can be found in various places. Examples being to give a false identity to the police to deter police investigations [17] or to misrepresent facts or people on social media for malicious purposes. The research at hand focuses on identity deception on big data platforms, like social media. A few examples of use cases of identity deception within social media are as follow:

- Influencing outcomes or results, like political campaigns [5]
- Enhancing or damaging the image of a company's brand [7, 8]
- Spam activity [15]
- Spreading fake news [4, 7]
- Pedophiles who lie about who they are to groom or approach a minor [3]

Current identity deception research, in social media platforms, cover a variety of use cases. The research to date however focus on fake accounts, or more commonly known as bots [8], which have been autogenerated for the indicated purpose. Current research has been found lacking for the following reasons:

- Very little focus has been placed on use cases where people are lying about who they are on social media platforms.
- Due to the scale of data on social media, human intervention is not plausible. An intelligent means is required to highlight potential threat.
- The attributes available on social media platforms are found to be lacking deceptive identification information.
- Attributes are treated equally. A method, like weighting, is required to enhance identity deception detection.
- The algorithms cannot be implemented real time to proactively predict and warn against deception based on the person's past behavior.

The current research project proposes to address these issues. The closest research found to date had success in showing potential identity deception when combining the name of the user with the color of the background account image [1] to understand if a person is lying about their gender. The results however could be limiting if the user has not updated their account's default image and color. The research at hand proposes a more holistic application of all plausible attributes towards an Identity Deception Indicator (IDI).

For the research at hand, the focus will be towards protecting minors on social media platforms. As mentioned earlier, pedophiles lie about their identity to gain the trust of a minor. Data, for the research, was gathered from Twitter as the social media platform. Only accounts pertaining to minors are evaluated together with their immediate network of friends and followers. The goal is to produce an algorithm that can find deceptive accounts within the mentioned corpus and address the points stated as lacking from previous research.

The research project has been divided into various methods discussed in more detail during previous research papers [16] and follows a scientific approach. The focus of this phase of the research, highlighted in green in Figure 1, was to use the identified deceptive attributes from the previous phase towards identifying deception. The results are discussed in section 3 of this report.

1.1 Main deliverables

The main deliverables of the past six months were:

- To enrich the corpus of data with deceptive features that can be used towards identity deception detection
- To produce a novel IDI per person
- To evaluate the results.
- To produce a ground truth set against which all future tests can be measured.

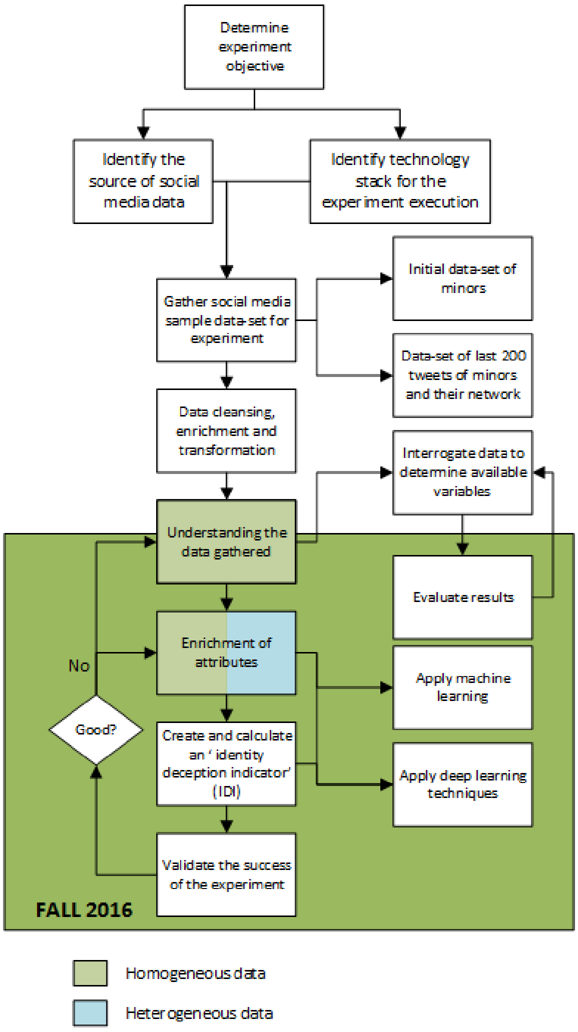

Determine experiment objective
Identify the source of social media data
Identify technology stack for the experiment execution
Gather social media sample data-set for experiment
Initial data-set of minors
Data-set of last 200 tweets of minors and their network
Data cleansing, enrichment and transformation
Interrogate data to determine available variables
Understanding the data gathered
Evaluate results
Enrichment of attributes
No
Apply machine learning
Good?
Create and calculate an ' identity deception indicator' (IDI)
Apply deep learning techniques
Validate the success of the experiment
FALL 2016
Homogeneous data
Heterogeneous data

Figure 1: The project process diagram

2 Use of HPI Future SOC Lab resources

To reiterate past feedback, the following resources were used for the research at the HPI Future SOC lab:

- Twitter: The Twitter4j Java API was used to dump the data needed for the experiment in a big data repository.
- Hortonworks Hadoop 2.4: For the purposes of this experiment HDP Hadoop runs on an Ubuntu Linux virtual machine hosted in "The HPI Future SOC"-research lab in Potsdam, Germany. This machine contains 4TBs of storage, 8GB RAM, 4 x Intel Xeon CPU E5-2620 @2GHz and 2 cores per CPU. Hadoop is well known for handling heterogeneous data in a low-cost distributed environment, which is a requirement for the experiment at hand.
- Flume: Flume is used as one of the services offered in Hadoop to stream initial Twitter data into Hadoop and into SAP HANA.
- Ambari: For administration of the Hadoop instance and starting/stopping the services like Flume.
- Java: Java is used to enrich the Twitter stream with additional information required for the experiment at hand and automate the data gathering process.
- SAP HANA: A SAP HANA instance is used which is hosted in "The HPI Future SOC"- research lab in Potsdam, Germany on a SUSE Linux operating system. The machine contains 4TBs of storage, 2TB of RAM (1.4TB effective) and 32CPUs / 100 cores. The in-memory high-performance processing capabilities of SAP HANA enables almost instantaneous results for analytics.

  The XS Engine from SAP HANA is used to accept streamed Tweets and populate the appropriate database tables.
- Machine learning APIs: Various tools are considered to perform classification, analysis and apply deep learning techniques on the data. These include the PAL library from SAP HANA, SciPy libraries in Python, Spark Mlib on Hadoop and the Hadoop Mahout service. For the research, R was the final choice. This decision was made due to support on this platform and libraries freely being available on the web community at a large scale.
- An additional Linux machine was provided for the lab to aid in the running of the CPU and memory intensive machine learning algorithms. The VM has 8 cores and 64GB of RAM.
- Visualization of the results will be performed by the libraries in R and PowerBI where appropriate.

The following ancillary tools were used as part of the experiment:

- For connection to the FSOC lab we used the OpenVPN GUI as suggested by the lab.
- For connecting and configuration of the Linux VM instance we used Putty and WinSCP.
- For connecting to the SAP HANA instance, we used SAP HANA Studio (Eclipse) 1.80.3.

3 Findings in the Fall 2016 semester

The purpose of this phase of the research project was to build a novel identity deception indicator (IDI) using some of the deceptive attributes identified in the previous phase. After initial results were achieved a second iteration were run towards improvement of the IDI.

3.1 First iteration

The following engineered features were identified towards an IDI:

- The friend, follower ratio [11]
- The type of images used as profile and background [14]
- The distance between the geo-location recorded on the SMP and the location as stated by the person [1]
- The sentiment, i.e. whether the overbearing language usage conveys a positive or negative feeling [7]
- The number of devices used [13]
- The timespan of activities on SMPs for a given person, compared to the corpus [12]

Each of these features were extracted per person and given a Deception Score (DS). A DS is the result of calculations used to ascertain a user's perceived deceptiveness, given the feature. The DS is defined as being in a range between 0 and 1, with 1 being more deceptive. The DS was calculated based on whether the feature was an outlier or not. The assumption is that outliers indicate a better likelihood of deception as most people are believed to be good and not to tell lies on social media platforms [2, 6]. Outliers are those data points that fall outside one of the following:

- Quartile 1 – (IQR x 1.5)
- Quartile 3 + (IQR x 1.5)

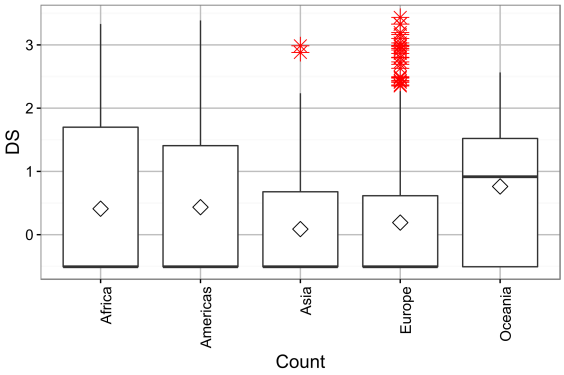


Figure 2: Outliers based on distance per continent

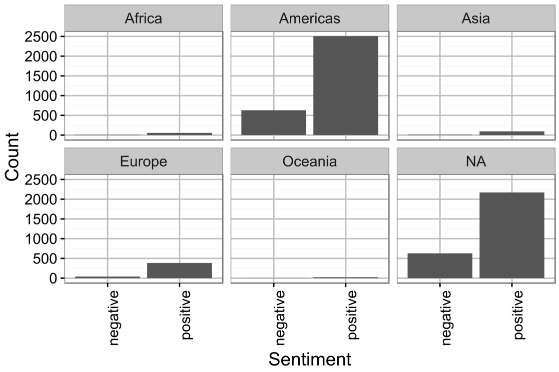


Figure 3: Overall sentiment per continent

where IQR refers to the Inter Quartile Range. (The above is also known as Tukey's method or Tukey's honest significance test [10])

Figure 2 shows the outliers, in red, based on geo location distance whereas Figure 3 shows initial results to determine that people with a negative sentiment are scarce and thus outliers.

A novel IDI was calculated by first adding all DS scores together for each person. The result was divided by the number of features (n) to get an average IDI per person. The formula can be depicted as:

$$IDI = (\sum_{k=1}^{n} DS(k))/n$$

The final step in the research experiment required the results to be evaluated. It was here where it was found, that there was no ground truth to compare the results against. The next iteration posed to address this shortfall.

Table 1: Determining a ground truth

Algorithm	AUC	Kappa	Recall
SVM (Radial)	0.958 965	0.473 266	0.324
Random Forest	0.966 967	0.817 629	0.868
J48	0.982 155	0.793 162	0.812
Bayes GLM	0.957 813	0.037 411	0.02
KNN	0.729 495	0.567 185	0.46
Adaboost	0.992 357	0.800 081	0.872
rpart	0.659 796	0.442 253	0.32
nnet	0.992 596	0.568 477	0.484

3.2 Second iteration

In this iteration, a dataset of 1,000 accounts was injected into the corpus containing only deceptive accounts. This dataset was manually generated and was proven to be representative of the original corpus.

The injected accounts were classified as deceptive whereas the original corpus was classified as being trustworthy. The introduction of this classification changed the approach for an IDI to being a binary classification problem. Binary classification problems can potentially be solved through supervised machine learning techniques. The experiment used 8 machine learning algorithms proven to have resolved binary classification problems on social media, like spam identification.

Only the original attributes from Twitter were used to get a relevant ground truth for following iteration comparisons. 155K+ accounts were used in the experiment using 5-fold, 3 repeat cross validation.

The results are shown in Table 1.

The results indicate that the UAC values are quite good but recall and Kappa are a reason for concern. The believe is that the UAC values are so good due to the highly-skewed nature of the classification. Kappa shows that almost half the algorithms are just guessing the correct answer.

The proposal is for more iterations to address these concerns. Only once a ground truth can be established, will we introduce the engineered features again to understand if the IDI can be improved or not via data enrichment.

3.3 Architecture

The SAP HANA instance, virtual machines and storage was provided by the HPI FSOC research lab and the following is worth mentioning:

- There were no issues in connection.
- The lab was always responsive and helpful in handling any queries.

- The environment is very powerful and more than enough resources are available which makes the HPI FSOC research lab facilities ideal for the experiment at hand
- Without the additional VM with more cores, we would not have been able to perform the machine learning computations.

Overall, we found that the environment and its power enabled the collection and handling of a big dataset without issue. The support of the HPI FSOC research lab is greatly appreciated.

4 Next steps for 2017

The deliverables for this phase are:

- To finalize the ground truth of results from supervised machine learning
- To analyze the effect of the skewness of data and determine actions based on these results
- To introduce the engineered features to the corpus
- To compare these results with the ground truth
- To improve the results by both catering for appropriately weighting certain features
- To introduce time as a variable
- To run the proposed model on real time data to identify outliers

References

[1] J. S. Alowibdi, U. A. Buy, S. Y. Philip, S. Ghani, and M. Mokbel. "Deception detection in Twitter". In: *Social Network Analysis and Mining* 5 (2015), pages 1–16. DOI: 10.1007/s13278-015-0273-1.

[2] M. D. Back, J. M. Stopfer, S. Vazire, S. Gaddis, S. C. Schmukle, B. Egloff, and S. D. Gosling. "Facebook Profiles Reflect Actual Personality, Not Self-Idealization". In: *Psychological Science* 21.3 (Jan. 2010), pages 372–374. DOI: 10.1177/0956797609360756.

[3] D. Bogdanova, P. Rosso, and T. Solorio. "Exploring high-level features for detecting cyberpedophilia". In: *Computer Speech & Language* 28 (2014), pages 108–120. DOI: 10.1016/j.csl.2013.04.007.

[4] C. Chen, K. Wu, V. Srinivasan, and X. Zhang. "Battling the internet water army: Detection of hidden paid posters". In: *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 2013, pages 116–120. DOI: 10.1145/2492517.2492637.

[5] N. J. Conroy, V. L. Rubin, and Y. Chen. *Automatic Deception Detection: Methods for Finding Fake*. 2015.

[6] L. Dedkova. "Stranger Is Not Always Danger: The Myth and Reality of Meetings with Online Strangers". In: *Living in the Digital Age. Self-Presentation, Networking, Playing, and Political Participation*. Edited by P. Lorentz, D. Smahel, M. Metykova, and M. F. Wrigh. muni press, Brno, 2015, page 78. ISBN: 978-80-210-7811-6.

[7] B. Drasch, J. Huber, S. Panz, and F. Probst. *Detecting Online Firestorms in Social Media*. 2015.

[8] S. Gurajala, J. S. White, B. Hudson, and J. N. Matthews. "Fake Twitter accounts: Profile characteristics obtained using an activity-based pattern detection approach". In: *Proceedings of the 2015 International Conference on Social Media & Society*. 2015, page 9. DOI: 10.1145/2789187.2789206.

[9] J. Hancock, J. Birnholtz, N. Bazarova, J. Guillory, J. Perlin, and B. Amos. "Butler lies: awareness, deception and design". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2009, pages 517–526. DOI: 10.1145/1518701.1518782.

[10] W. C. Navidi. *Statistics for engineers and scientists*. McGraw-Hill New York, 2010. ISBN: 978-0-07-337633-2.

[11] D. Quercia, M. Kosinski, D. Stillwell, and J. Crowcroft. "Our Twitter profiles, our selves: Predicting personality with Twitter". In: *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom)*. 2011, pages 180–185. DOI: 10.1109/PASSAT/SocialCom.2011.26.

[12] N. M. Radziwill and M. C. Benton. *Bot or Not? Deciphering Time Maps for Tweet Interarrivals*. 2016. arXiv: 1605.06555 [cs.SI].

[13] D. Robinson. "Two people write Trump's tweets. He writes the angrier ones". In: *Washington Post* (2016).

[14] S. Sharma. "Black Twitter? Racial hashtags, networks and contagion". In: *New Formations* 78 (2013), pages 46–64. DOI: 10.3898/NewF.78.02.2013.

[15] S. J. Soman and S. Murugappan. "Detecting malicious tweets in trending topics using clustering and classification". In: *Recent Trends in Information Technology (ICRTIT)*. 2014, pages 1–6. DOI: 10.1109/ICRTIT.2014.6996188.

[16] E. V. der Walt and J. H. P. Eloff. "Protecting minors on social media platforms - A Big Data Science experiment". presented at the HPI Cloud Symposium "Operating the Cloud", Potsdam, Germany. 2015.

[17] G. Wang, H. Chen, and H. Atabakhsh. "Criminal identity deception and deception detection in law enforcement". In: *Group Decision and Negotiation* 13 (2004), pages 111–127. DOI: 10.1023/B:GRUP.0000021838.66662.0c.

Global-Scale Internet Graphs

Vulnerability Analysis Based on Worm-Spread Simulations

Benjamin Fabian[1,2], Annika Baumann[1], Tatiana Ermakova[3], and Stefan Kelkel[1]

[1] Chair of Information Systems
Humboldt-Universität zu Berlin
{bfabian,annika.baumann}@wiwi.hu-berlin.de
stefan.kelkel@googlemail.com
[2] HfT Leipzig
[3] University of Potsdam
tatiana.ermakova@uni-potsdam.de

Based on our traceroute data integrated from global-scale mapping projects aimed to generate comprehensive Internet maps at different abstraction levels, we analyze the Internet graph in terms of important nodes. In an evolution of the initial project, we start to assess several malware strategies that could affect border routers. We present some preliminary results on the communication ability in the presence of different numbers of disconnected routers.

1 Introduction

Our prior research project [13] aimed at developing methods for creating and analyzing a large integrated set of Internet graphs at several abstraction levels.

This serves as a basis for our ongoing analyses searching for bottlenecks and weak points in the entire Internet topology as well as in the topological connectivity of individual firms and services.

As our project evolved, a novel line of research studies attack strategies motivated by autonomic malware that could affect important border routers, and investigates their impact on Internet robustness via graph-based simulations.

2 Research Approach

We simulated systematically destructive Internet worms that would affect border routers and began to study their effects.

3 Related Publications

The Institute of Information Systems at Humboldt University Berlin has been conducting research based on graph analysis for several years [1, 2, 3, 4, 5, 6, 11, 13, 15, 16].

In particular, robustness analyses and vulnerability assessments of the Internet at the AS-level have been conducted. Large-scale graph analysis has also been applied on the Bitcoin transaction network [6, 16] and Twitter use in the political sphere [5].

Further publications based on the current project are under development[8, 9, 10, 12]. We list some of them as white papers but note that some of them are not finalized while others are currently under review.

Our core framework CORIA for making the results of such graph analyses useful in practical applications will be presented at IEEE ICC 2017 in Paris [9].

Aspects of our research have also been covered in a major German newspaper, "*Der Tagesspiegel*" [7] and "El Espanol" [17].

4 Project Plan

Our project requires powerful computation capabilities based on the large-scale memory and multicore architecture of the HP Converged Cloud and, potentially, also the newly implemented SAP HANA Graph Engine.

This project is structured in several phases. The current phases of our project study the destructive capabilities of malware spreads via outage simulations.

With the help of the computational power of HPI Future SOC Lab, we are much better equipped to examine graph measures such as centrality metrics, clustering coefficients, shortest paths, and connected components.

The novel project phases will also be carried out on the resources provided by the HPI Future SOC Lab [14].

5 Project Status and Results

5.1 Simulations of Attacks Inspired by Malware

As previously stated, we will simulate attacks inspired by malware threats. A systematic vulnerability assessment of the Internet's core components would help to identify critical areas and to build more resilient structures.

In particular, the border routers, which are interconnecting the autonomous systems, could constitute critical bottlenecks.

This work phase attempts to quantify the impact of attacking border routers at the autonomous system (AS) level.

For this purpose, various worms are simulated. These worms are able to infiltrate the router operating systems of leading suppliers, such as Cisco and Juniper.

In order to identify these router operating systems, a TTL-based fingerprinting method is conducted.

The results for the different attack scenarios will be compared and investigated.

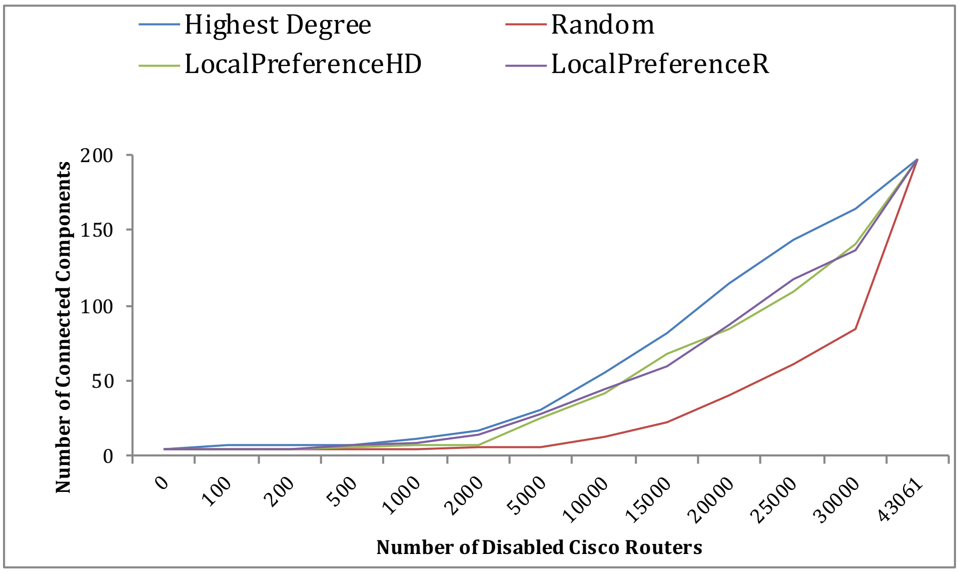


Figure 1: Number of Connected Components

5.2 Preliminary Results

We will focus on the so-called Ark ITDK AS graph, which will be discussed in [10]. This graph has 38,417 nodes and more than 230,000 edges.

We observe that this graph exhibits *Small World* characteristics, such as a small average shortest path length of 3.4.

An example of the simulations we conduct is shown in Figure 1.

Here, four AS-eclipsing spreading strategies are studied with respect to their impact on the number of disjoint connected components of the entire graph, an indicator of the (in-)ability of network regions to still communicate globally after an attack.

Further details will be presented in [10].

5.3 Use of Hardware Resources

The hardware provided by the HPI Future SOC Lab so far included three HP Converged Cloud Blades with 24 x 64-bit CPUs running at a frequency of 1.2 GHz on Ubuntu 14.04. Each of the three machines had 64 GiB of memory and was equipped with 1 TiB HDD. This configuration offers an extensive parallelization of tasks.

The calculated results would not have been possible without the support of the HPI Future SOC Lab [14]. Due to the intensive calculations necessary for the vulnerability analyses, the use of HPI Future SOC Lab resources has been very important for years. We are very grateful for the continuing support.

6 Conclusion

In future work, we aim to conduct further vulnerability analyses based on the extensive data sets that we have collected so far.

Specifically, we are going to apply a variety of further graph metrics and simulate attack strategies, both requiring massive calculation capacities enabled through HPI Future SOC Lab resources.

References

[1] A. Baumann and B. Fabian. "How Robust is the Internet? – Insights from Graph Analysis". In: *Proceedings of the 9th International Conference on Risks and Security of Internet and Systems (CRiSIS 2014), Trento, Italy, Springer, LNCS 8924*. 2014. DOI: 10.1007/978-3-319-17127-2_18.

[2] A. Baumann and B. Fabian. "Towards Measuring the Geographic and Political Resilience of the Internet". In: *International Journal of Networking and Virtual Organisations* 13.4 (2013), pages 365–384. DOI: 10.1504/IJNVO.2013.064465.

[3] A. Baumann and B. Fabian. "Vulnerability Against Internet Disruptions – A Graph-based Perspective". In: *Proceedings of the 10th International Conference on Critical Information Infrastructures Security (CRITIS 2015), Berlin, Germany*. Oct. 2015. DOI: 10.1007/978-3-319-33331-1_10.

[4] A. Baumann and B. Fabian. "Who Runs the Internet? Classifying Autonomous Systems into Industries". In: *Proceedings of the 10th International Conference on Web Information Systems and Technologies (WEBIST), Barcelona, Spain*. Apr. 2014.

[5] A. Baumann, B. Fabian, S. Lessmann, and L. Holzberg. "Twitter and the Political Landscape – A Graph Analysis of German Politicians". In: *Proceedings 24th European Conference on Information Systems (ECIS 2016), Istanbul, Turkey*.

[6] A. Baumann, B. Fabian, and M. Lischke. "Exploring the Bitcoin Network". In: *Proceedings of the 10th International Conference on Web Information Systems and Technologies (WEBIST 2014)*. 2014, pages 369–374.

[7] R. Cisielski. *Schatz, das Internet ist kaputt: Immer mehr Menschen nutzen das Netz immer mehr Maschinen und Prozesse sind davon abhängig. Doch was passiert, wenn es zur digitalen Apokalypse kommt? In Berlin simulieren Wissenschaftler den Totalausfall*. Der Tagesspiegel. Sept. 24, 2016. URL: https://blendle.com/i/der-tagesspiegel/schatz-das-internet-ist-kaputt/bnl-tagesspiegel-20160924-0011949942 (last accessed 2017-03-27).

[8] B. Fabian, A. Baumann, S. Dombrowski, and T. Ermakova. *Towards Graph-Based Simulations of Cloud Connectivity*. in submission. 2017.

[9] B. Fabian, A. Baumann, M. Ehlert, T. Ermakova, and V. Ververis. "CORIA – Analyzing Internet Connectivity Risks Using Network Graphs". In: *Proceedings IEEE International Conference on Communications (ICC 2017), Paris, France*. 2017. DOI: 10.1109/ICC.2017.7996828.

[10] B. Fabian, A. Baumann, T. Ermakova, and S. Kelkel. *Internet Robustness Analysis – Simulation of Worm-Based Router Attacks*. Working Paper. 2017.

[11] B. Fabian, A. Baumann, and J. Lackner. "Topological Analysis of Cloud Service Connectivity". In: *Computers & Industrial Engineering* 88 (Oct. 2015), pages 151–165. DOI: 10.1016/j.cie.2015.06.009.

[12] B. Fabian, A. Baumann, G. Tilch, and T. Ermakova. *A Snapshot of the Internet Topology*. Working Paper. 2017.

[13] B. Fabian and G. Tilch. *Analyzing the Global-Scale Internet Graph at Different Topology Levels: Data Collection and Integration*. HPI Future SOC Lab Day Workshop & Report. 2015.

[14] *HPI Future SOC Lab*. URL: https://hpi.de/forschung/future-soc-lab.html (last accessed 2017-03-27).

[15] M. Huth and B. Fabian. "Inferring Business Relationships in the Internet Backbone". In: *International Journal of Networking and Virtual Organisations* (2016). DOI: 10.1504/IJNVO.2016.081651.

[16] M. Lischke and B. Fabian. "Analyzing the Bitcoin Network: The First Four Years". In: *Future Internet* 8 (1 Mar. 2016). DOI: 10.3390/fi8010007.

[17] S. Martínez. *El coste económico de un apocalipsis en internet*. El Espanol. Oct. 14, 2016. URL: http://www.elespanol.com/economia/20161014/162984533_0.html (last accessed 2017-03-27).

Multimodal Recurrent Neural Network for Generating Image Captions with multiple Granularity

Yash Choudhary and Thilini Cooray

Singapore University of Technology and Design
f2013678@goa.bits-pilani.ac.in,muthuthanthringe@mymail.sutd.edu.sg

Language generation on images is a fundamental problem in artificial intelligence that connects computer vision and natural language processing. We continued our previous experiments of image captioning using a multimodal recurrent neural network and analyzed its behavior based on the granularity level of information fed to the network during training. Several approaches have been suggested for image captioning. Some of them give great results but sometimes they are much complex and not flexible. The image captioning model that we adopted is comparably simple and can provide satisfying results. It consists of a convolutional neural network and a bidirectional recurrent neural network. All our experiments were carried out using Flickr8k dataset. Apart from evaluating the model with the original captions from dataset, we also generated a caption dataset removing color attributes and analyzed the results of new model. All evaluation results are included in this report.

1 Introduction

Humans can describe any visual scene they come across using natural language within milliseconds regardless of its complexity. They can detect most significant parts of images or videos and build connections among those objects very quickly. Human brain is functioning in an amazing way to do these complex tasks highly efficiently. Is it possible to implement brain's mechanism of understanding visuals and describing them in a machine? This has been a highly-discussed domain in computer vision for a long time and recently, research on this improved significantly with the advancement of artificial intelligence. We continued our experiments started last term on language generation from images. We adopted an existing image captioning model from the literature and modified it to cater our research direction. This model is capable of providing complete captions from natural language when an image is fed to it.

Generating descriptions for images is a very challenging task. However, it could be highly useful for many fields. It specially can give significant impact on computer vision applications. Visually impaired people can receive many benefits from systems which can summarize visual scenes in natural language. It will help them to understand their surroundings better and act accordingly. Image captioning is significantly complex compared to well-studied image classification or object recognition. In this task, the model should generate description not only about objects in the image, but

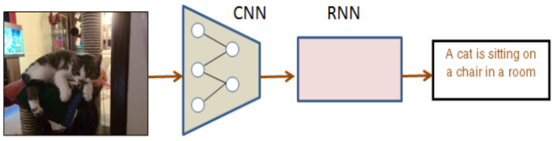


Figure 1: Image Caption Generating Model

it also must describe their relationships with each other, their attributes and the activities they are involved in. Also to present image captions in a human understandable way, the model should manipulate natural language processing aspects as well.

Computer vision and natural language processing (NLP) has been researched as two separate fields for decades and both domains have achieved huge advances on theirs. However, when it comes to generate language using images, both fields need to be bridged together. Once an image is fed to a captioning model, the model should be able to detect most interesting areas of that image as the first step. This involves computer vision; object detection to be more precise. Once these detections are done, a language model should be able to understand a computer vision based interpretation of the image and provide a natural language description to it. This is beyond the general task of sentence generation in NLP.

The main model used in our experiments consists with two sub models which are Deep Convolutional Neural Network (CNN) which is used for converting images to feature vectors and Multimodal Recurrent Neural Network (RNN) for caption generation.

In the last few years object detection and feature recognition enhanced rapidly in machine learning field and the hardware capabilities of computers also improved hugely proving us with highly effective processing solutions such as Graphical Processing Units (GPU). As a result, deep learning techniques which can scan through billions of data records such as CNNs evolved to produce rich representations of the input images by embedding images to fixed length vectors. These vectors represent any features of the image and it can be used to so many computer vision tasks. In this model, CNN is used to represent the image and give the features of the image as the output. These feature vectors can be used as input to the multimodal recurrent neural network to generate natural language descriptions. When we give features of the image which was received from CNN to the RNN, it generates most probable language description for the image considering its feature vector. CNN and RNN are used to develop a joint model for image caption generation. (See Figure 1)

Several models have been suggested for image captioning tasks in past few years and we continue our experiments with the model introduced by Andrej Karpathy and Li Fei-Fei [4]. In their paper, Karpathy et al. [4] describe an image captioning model which can output both single sentence image captions and dense captions for each area of interest. However, we only use the single sentence image captioning capability of this model.

After implementing the image captioning model, it can be used for many different applications. Apart from experimenting with available image captioning data, we started evaluating the model with different levels of granularity in detail of captions. In this report, we present our results on training the model with image captions where attributes describing object colors have removed.

2 Model

As shown in Figure 1, image caption generating model has two main parts. Using both of them we can generate the description for an image. First part is the CNN model which encodes the given image to a fixed size feature vector. Then the output of the CNN model is given to the RNN model as an input. Bidirectional RNN model then outputs a sentence describing the image. To get accurate results we have to optimize these two models.

2.1 Convolutional neural network for feature detection of image

We did the experiment following Karpathy et al. [4]. Authors have used a pre-trained model for CNN as their main focus is on language generation rather than optimizing the CNN. Therefore features of images using a pre-trained VGG Net model, that is one of the improved versions of the models used by the VGG team in the ILSVRC-2014 competition their CNN was implemented. These models are developed by following the work of Karen Simonyan et al. [9] about very deep convolution networks.

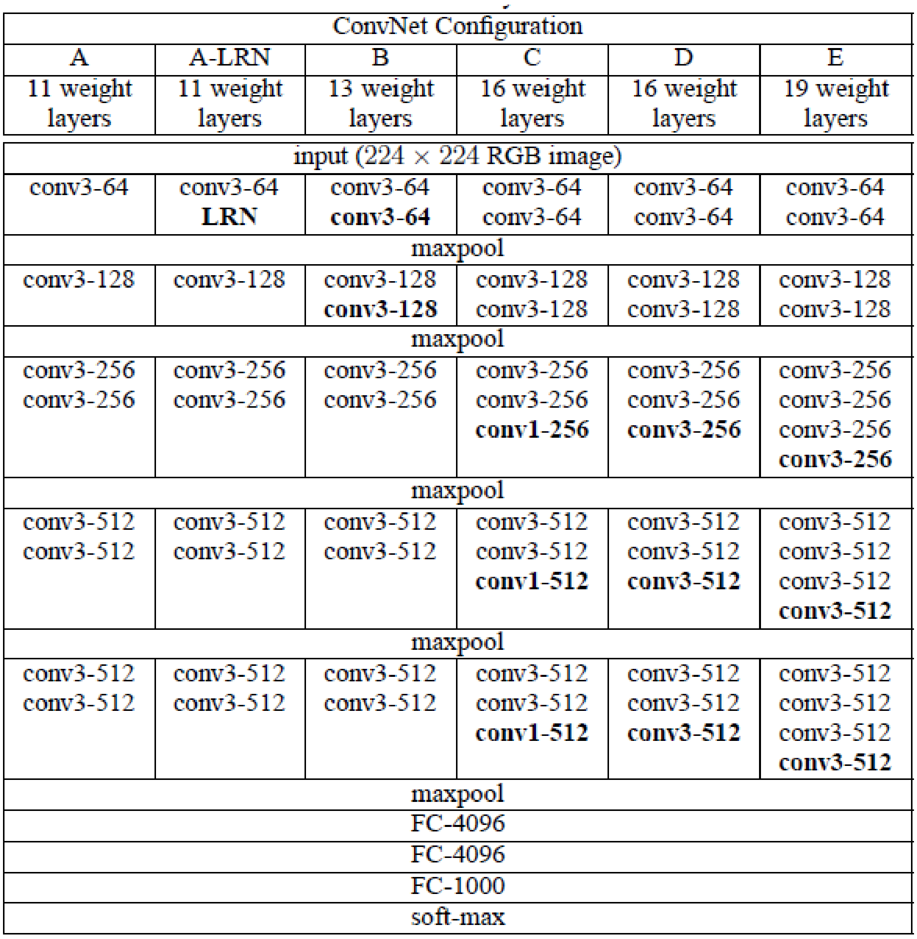
For experiments, we used 16 layer VGG Net as we were able to find out that 19 layer version also provides very similar results at the cost of performance.

Let's consider all configurations of convolution neural networks which are described in Karen Simonyan et al. [9] paper. (Figure 1)

During training and testing, the input to this convolution neural network is a fixed size 224 x 224 RGB image. The only preprocessing done in the model is subtracting the mean RGB value computed on the training set, from each pixel. After preprocessing the image, it is passed through a stack of convolutional layers which has filters with a very small receptive field: 3x3 (which is the smallest size to capture the notion of left/right, up/down, center). In one of the configurations they also utilize 1×1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of convolution layer input is such that the spatial resolution is preserved after convolution, i.e. the padding is 1 pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2. [9]

A stack of convolutional layers (which has a different depth in different architectures) is followed by three Fully-Connected (FC) layers: the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains

Table 1: ConvNet configurations of CNNs [9]



ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks. However for this image captioning model, we do not want image classification and we only want features of the image which we can give to the RNN model as an input. The multimodal recurrent neural network which developed by following Karpathy et al. [4] is implanted to get image feature input as a 4096 dimension vector which gives layer before last Fully-connected layer in VGG Net. All hidden layers are equipped with the rectification (ReLU (Krizhevsky et al., 2012 [5])) non-linearity.

In these experiments, we used configuration D (in Figure 1) pre-trained convolutional neural network for image feature detection with neglecting last two layers of the configuration. Therefore, we got output as a 4096 dimension feature vector. This vector will then be sent through a linear layer and encode it to 512 followed by another ReLU layer. This feature vector can give to the RNN model as an input to generate a sentence that describes the image.

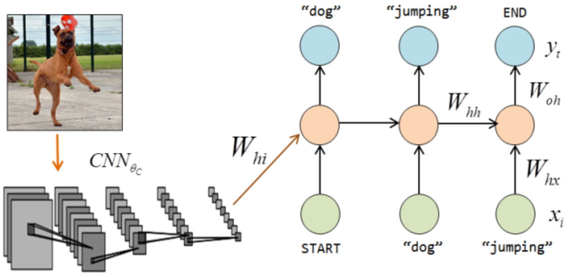


Figure 2: Recurrent neural network

2.2 Multimodal recurrent neural network for generating descriptions

The section describes the multimodal neural network developed by Karpathy et al. [4] for sentence generating for a given image input. The key challenge is the design of a model that can predict a variable-sized sequence of words for a given image. In previously developed language models based on Recurrent Neural Networks (RNNs) [2, 7, 10], this is achieved by defining a probability distribution of the next word in a sequence given the current word and context from previous time steps. In the paper, they have introduced a simple but effective extension that additionally conditions the generative process on the content of an input image. More formally, during training the Multimodal RNN takes the image pixels I and a sequence of input vectors $(x_1,..., x_T)$. It then computes a sequence of hidden states $(h_1,..., h_T)$ and a sequence of outputs $(y_1,..., y_T)$ by iterating the following recurrence relation for $t = 1$ to T. [4]

$$
\begin{aligned}
b_v &= W_{hi}[CNN_{\theta_C}(I)] \\
h_t &= f(W_{hx}x_t + W_{hh}h_{t-1} + b_h + \mathrm{I}(t = 1)\mathrm{e}\; b_v) \\
y_t &= softmax(W_{oh}h_t + b_o)
\end{aligned}
$$

In the equations above, W_{hi}, W_{hx}, W_{hh}, W_{oh}, x_i and b_h, b_o are learnable parameters, and $CNN_{\theta_C}(I)$is the last layer of a CNN model. x_i is a representation of each word in vocabulary and it can be create using word embedding method. The output vector y_t holds the (unnormalized) log probabilities of words in the dictionary and one additional dimension for a special END token. Note that they provide the image context vector b_v to the RNN only at the first iteration, which Karpathy et al. [4] found to work better than at each time step. They also found that it can help also to pass both b_v, $(W_{hx}x_t)$ through the activation function. A typical size of the hidden layer of the RNN is 512 neurons. (Refer to Figure 2)

After implementing the above model, we can train the model using input data set with images and corresponding image describing sentences. Training process is descried in Kaparthy et al. [4].

The RNN is trained to combine a word (x_t), the previous context (h_{t-1}) to predict the next word (y_t). During the training process, they condition the RNN's predictions on the image information (b_v) via bias interactions on the first step. The training proceeds as follows (refer to Figure 2): First, set $h_o = 0, x_1$ to a special START vector, and the desired label y_1 as the first word in the sequence. Analogously, set x_2 to the word vector of the first word and expect the network to predict the second word, etc. Finally, on the last step when x_T represents the last word, the target label is set to a special END token. The cost function is to maximize the log probability assigned to the target labels (i.e. Softmax classifier). Using cost function and train the model with cost minimization we can reduce the error of the prediction over the time. When we train the model parameters get update and they train to give accurate result. To get more accurate results we have to train the model on larger datasets.

In the test time, we can input an image feature vector to the trained RNN model and generate sentence for corresponding image. To predict the sentence, first we have to create image representation b_v, set $h_o = 0, x_1$ to the START vector and compute the distribution over the first word y_1. Then we can sample a word from the distribution (or pick the argmax), set its embedding vector as x_2, and repeat this process until the END token is generated. After generating the END token we can get the well-structured sentence as a output from the RNN model.

3 Experiments

3.1 Datasets

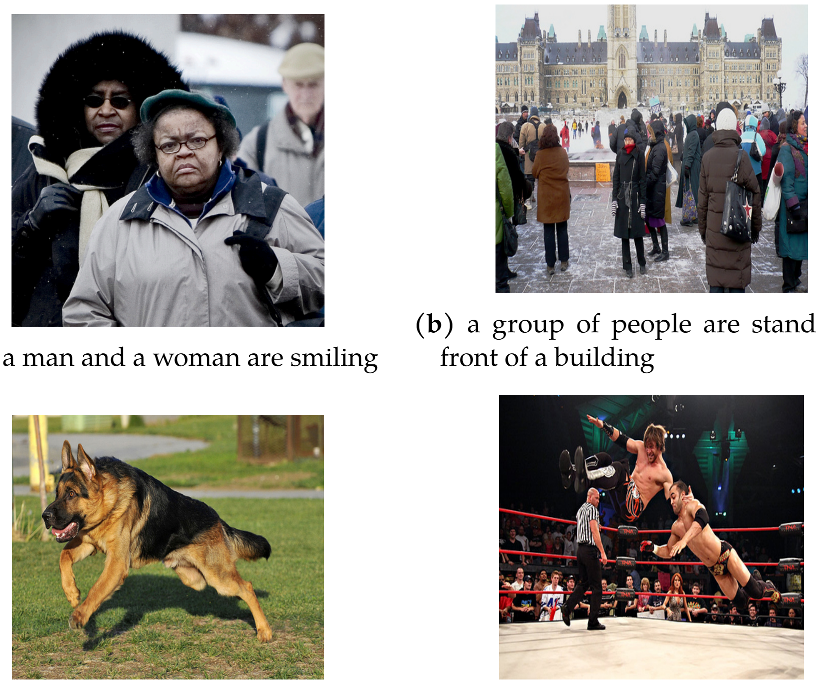
In this experiments, we have used Flickr8K [3] dataset. This dataset contains 8,000 images and each is annotated with 5 sentences using Amazon Mechanical Turk [4]. For Flickr8K, we allocated 1000 images for testing, 1000 images for validation and rest of the images for training.

3.2 Implementation

We used the CNN-RNN joint model implemented by Kaparthy et al. [4] after making suitable changes to execute this image caption generator. They have implemented the model using Torch. In this model they have presented model training and image caption prediction source code. It was instructed by authors to use VGG Net 16 layer pre-trained model for the CNN. Therefore during the training phase, this model mainly refines its RNN. We trained the model on a GPU. We enabled parameter decaying and language model based validation during training phase.

During the process of the training we collected some trained model checkpoints over the time. Then we evaluated each of their outcomes and compared their accuracy. All the results are discussed in the next section.

After training the model over dataset, we tried to test the accuracy of the model using some test images and comparing their natural language descriptions manually. Example sentences generated by multimodal recurrent network for some test images are shown below in Figure 3. (Other example images with categorization for accuracy are in Appendix A). As shown in the Figure 3, some of the captions have described



(a) a man and a woman are smiling

(b) a group of people are standing in front of a building

(c) a black and brown dog runs through the grass

(d) a basketball player in white uniform is trying to shoot a basket

Figure 3: Image captions for example. Accurate: b and c. Partially correct: a. Inaccurate: d.

the image accurately, some descriptions are partially correct and some of them have incorrect description for given image. However most of the generated descriptions are correct and we can improve the accuracy by training the model with large amount of training data.

3.3 Results and Evaluation

This section presents results we achieved through the model on 1000 test images from Flickr8k dataset.

After training the model using 6000 images from Flickr8K dataset with validation step with 1000 images after every 2500 iterations during training, we started testing. During the training process, which continued for around 2 days, we saved several checkpoints of the training model after precise number of iterations and then we tested each training model checkpoint to measure its accuracy.

Each generated image description from the model in the testing phase evaluated on aspects like cross entropy loss and BLEU [8] score, CIDEr [11], ROUGE [6] and METEOR [1].

All the trained model checkpoints and their evaluation aspects are shown in the Figure 2. (B-n is BLEU score that uses up to n-grams. High is good in all columns). Each trained model checkpoints are saved in each time period cross validations are

Table 2: BLEU score evaluation

(**a**) BLEU score evaluation for image captioning

Number of Iteration	Bleu 1	Bleu 2	Bleu 3	Bleu 4	METEOR	ROUGE	CIDEr	Loss
2500	0.519	0.329	0.209	0.138	0.161	0.389	0.267	3.170
5000	0.519	0.329	0.209	0.138	0.161	0.389	0.267	3.170
7500	0.512	0.324	0.207	0.135	0.159	0.392	0.286	3.271
10 000	0.512	0.324	0.207	0.135	0.159	0.392	0.286	3.271
12 500	0.512	0.324	0.207	0.135	0.159	0.392	0.286	3.271
15 000	0.512	0.324	0.207	0.135	0.159	0.392	0.286	3.271
17 500	0.526	0.346	0.227	0.150	0.174	0.407	0.345	3.731
20 000	0.528	0.345	0.223	0.145	0.178	0.410	0.345	3.840
22 500	0.542	0.359	0.236	0.154	0.178	0.413	0.365	3.891
25 000	0.547	0.364	0.240	0.157	0.182	0.416	0.374	3.883
50 000	0.539	0.355	0.231	0.150	0.181	0.415	0.359	4.079
100 000	0.538	0.353	0.229	0.149	0.179	0.415	0.359	4.075

(**b**) BLEU score evaluation for image captioning without colors

Number of Iteration	Bleu 1	Bleu 2	Bleu 3	Bleu 4	METEOR	ROUGE	CIDEr	Loss
2500	0.545	0.328	0.188	0.115	0.150	0.385	0.225	3.147
5000	0.545	0.328	0.188	0.115	0.150	0.385	0.225	3.147
7500	0.540	0.342	0.206	0.126	0.160	0.399	0.271	3.219
10 000	0.540	0.342	0.206	0.126	0.160	0.399	0.271	3.219
12 500	0.526	0.325	0.195	0.122	0.154	0.387	0.268	3.519
15 000	0.526	0.325	0.195	0.122	0.154	0.387	0.268	3.519
17 500	0.531	0.334	0.204	0.126	0.162	0.399	0.279	3.680
20 000	0.548	0.348	0.212	0.130	0.170	0.406	0.334	3.768
22 500	0.552	0.355	0.220	0.138	0.169	0.414	0.331	3.899
25 000	0.555	0.351	0.218	0.137	0.167	0.410	0.314	3.913
50 000	0.545	0.350	0.214	0.133	0.167	0.410	0.314	3.983
100 000	0.551	0.355	0.214	0.131	0.168	0.409	0.316	3.974

done. When the checkpoint number increase that says it has trained much more number of iterations or number of training images. That's mean when we go down through the table the time and number of iterations the model has trained is increase. Therefore, as we can see the BLEU score of the each checkpoint is getting increase when we go down the table. That means if we can give more time and number of iterations to train the model we would get more accurate results from the RNN model.

3.4 Analysis of the trained model after changing the granularity of information in caption data

After training and testing the model, we have done some experiments using image caption generator. As an application of the model, we tried to analyze the difference in performance by adjusting the parameters of training set using this model.

While analyzing the image caption dataset, we observed that it contains captions with different levels of information. For an example, let's take a sample image with two dogs playing. There were some descriptions which basically mentioned "two dogs are playing". Some have added more information by saying "A big dog and a middle-sized dog in playing on a grassy land". Some have further added details such as "A brown dog is playing with a black and white dog on a field". While analyzing the data, we found out that most people have used color attribute of objects when further emphasizing their observations. And also as RNN is generating word sequences considering the tokens which maximizes the probability of whole sentences when that token is added, RNN tend to give similar types of sentences in every place where same color is used. For an example, if it describes a shirt, it always tends to output blue or red despite its real color. This is because it is more probable to have term "red" or "blue" before shirt in the training set. On the other hand, as per our observations, when RNN caught such additional attribute observation from image feature vector, it seems to be making the image caption around that attribute ignoring all other more important aspects which needs to be captured in the description.

Therefore, we tried to analyze the behavior of RNN when it is trained with caption data which does not have color attribute. We created a separate dataset removing colors from all the places where it is used to describe an object. Then we trained the RNN using this dataset. Results of this experiment is in Figure 2(b). We found out that this new model is capable of describing more areas with interest in an image compared to previous model.

4 Future Work

Even though we were able to replicate results of the original paper [4], it would be interesting to further fine tune the model and check whether we can further enhance accuracies. At the moment, we have only tested on model behavior after removing color attribute of the captions. It will be an interesting study to further consider models with different levels of information and combine them.

5 Conclusion

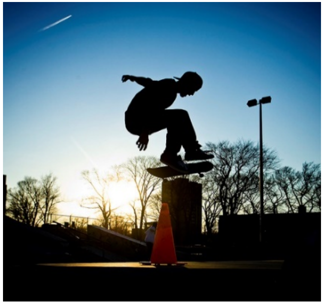
Automatic image captioning is a field which recently drew attention of researchers from both computer vision and NLP. Several approaches have been suggested to achieve this goal and we also tried to experiment on image captioning models in order to understand their behaviors and identify further improvements which can be achieved through them. We adopted an existing model presented by Karpathy et al. [4] as our starting point and continued our modifications from there. After replicating results of the original paper, we looked for further experiments and turned to analyze behaviors of models when the granularity of information in the caption are changed. Currently we evaluated a model which was trained removing color attributes of captions. We are planning to carry out further experiments on this direction.

Acknowledgement

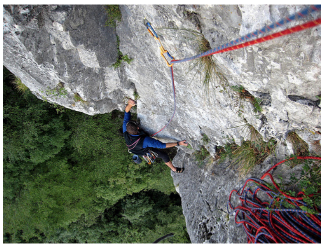
We are grateful to Future SOC Group for providing us with required computational power including GPUs and Servers for conducting our experiments.

A Image captions generated by Model 1

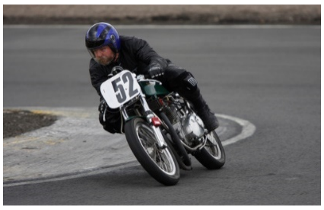
Good Results



a man jumps of a ramp on his skateboard

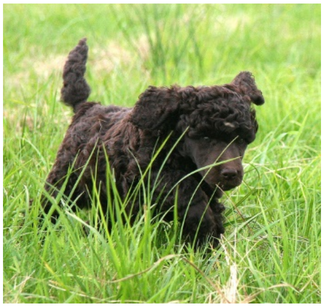


a man is climbing a rock wall

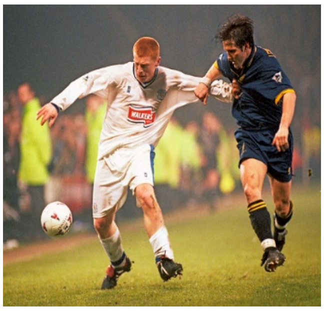


a motorcycle racer is turning on a race track

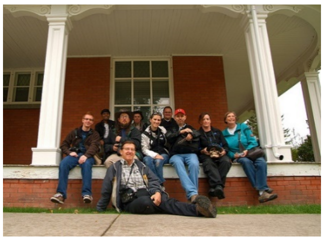
Moderate results



a black dog is playing with a red ball

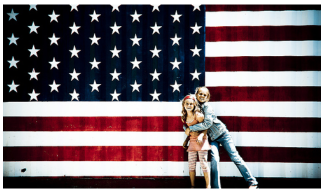


a man in blue uniform kicks a soccer ball on a field

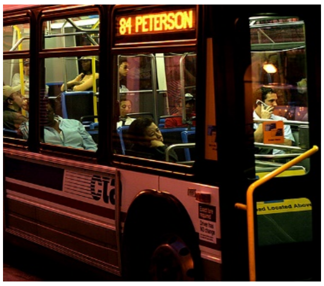


a group of people are standing in a line in front of a white house

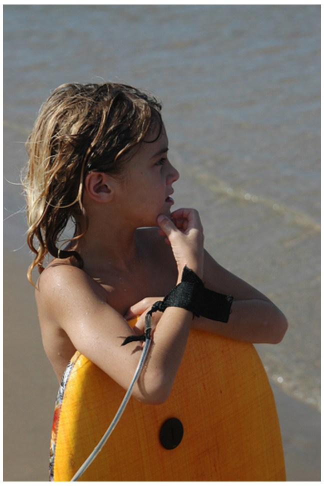
Unacceptable results



A woman in a red jacket is sitting on a bench



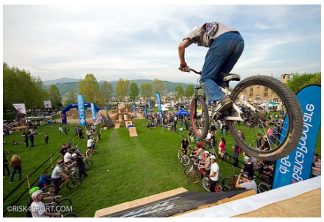
a man sits on a bench in a park



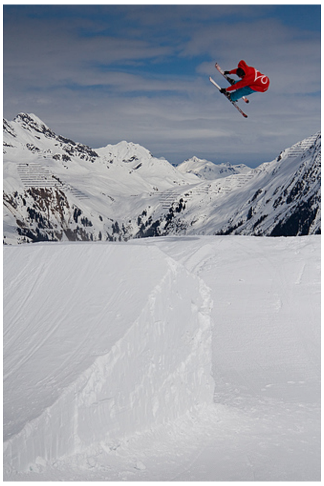
a young girl wearing a pink hat and sunglasses smiles

B Image captions generated by Model 2

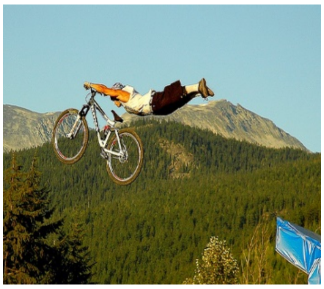
Good results



a cyclist is performing a jump on a bicycle

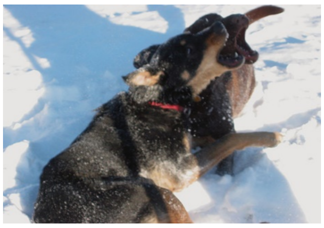


a skier in a jacket and helmet is skiing down a hill

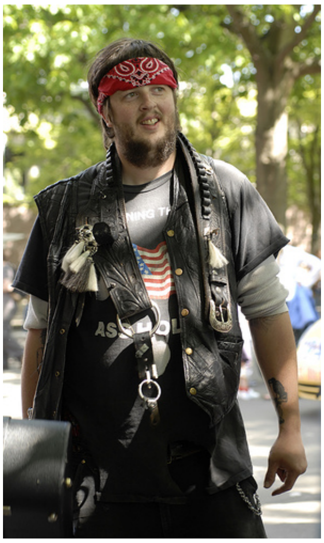


a man in a helmet is riding his bike on a ramp

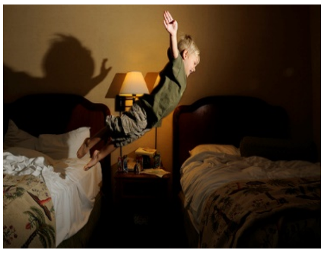
Moderate Results



a dog is standing in the snow

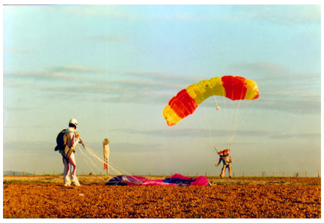


a man with a hat and sunglasses is smiling

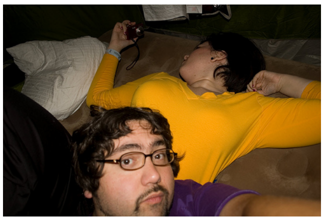


a person is sitting on a couch and knocks over a lamp

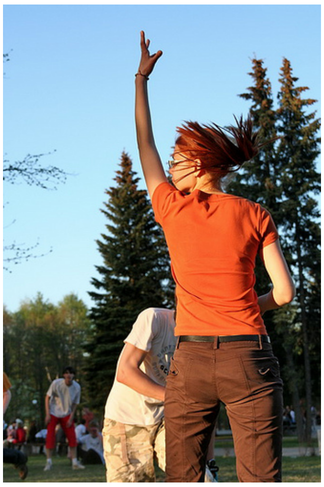
Unacceptable Results



two people are on a beach in front of a large wave

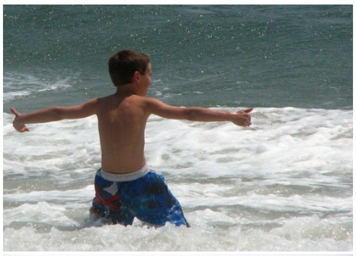


a man with a tattoo on his arm cooking something in a frying pan

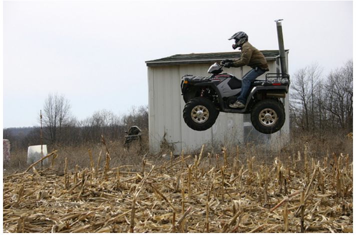


a girl in a shirt and pants is jumping over a fence

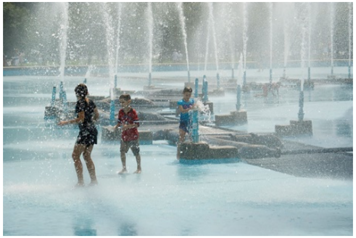
C Comparison of Image captions generated by Model 1 and Model 2.



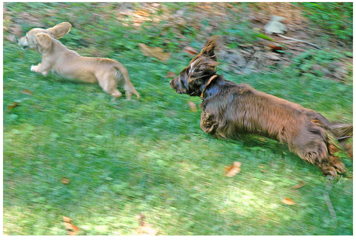
Model 1: a young boy in a blue bathing suit jumps off of a dock into a lake
Model 2: a boy in a swim trunks is standing in a pool of water



Model 1: a person is riding a yellow atv over the gate
Model 2: a person is riding a bike in the woods



Model 1: a group of people are standing on a beach
Model 2: a group of people are in the air above the water



Model 1: a brown dog is running in the grass
Model 2: two dogs are playing together in the grass

References

[1] S. Banerjee and A. Lavie. “METEOR: An automatic metric for MT evaluation with improved correlation with human judgments”. In: *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. Volume 29. 2005.

[2] J. L. Elman. “Finding structure in time”. In: *Cognitive science* 14.2 (1990), pages 179–211. DOI: 10.1207/s15516709cog1402_1.

[3] M. Hodosh, P. Young, and J. Hockenmaier. “Framing image description as a ranking task: data, models and evaluation metrics”. In: *Journal of Artificial Intelligence Research* (2013). DOI: 10.1613/jair.3994.

[4] A. Karpathy and L. Fei-Fei. *Deep visual-semantic Alignments for generating image descriptions*. 2014. DOI: 10.1109/TPAMI.2016.2598339. arXiv: 1412.2306.

[5] A. Krizhevsky, L. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks". In: *Advances in Neural Information Processing Systems 25 (NIPS 2012)*. 2012. DOI: 10.1145/3065386.

[6] C.-Y. Lin. "Rouge: A package for automatic evaluation of summaries". In: *Text summarization branches out: Proceedings of the ACL-04 workshop*. Volume 8. 2004.

[7] T. Mikolov, M. Karafi´at, L. Burget, J. Cernock'y, and S. Khudanpur. "Recurrent neural network based language model". In: *INTERSPEECH*. 2010.

[8] K. Papineni, S. Roukos, T. Ward, and W. Zhu. "Bleu: a method for automatic evaluation of machine translation". In: *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pages 311–318. DOI: 10.3115/1073083.1073135.

[9] K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition*. 2014. arXiv: 1409.1556.

[10] I. Sutskever, J. Martens, and G. E. Hinton. "Generating text with recurrent neural networks". In: *ICML*. 2011.

[11] R. Vedantam, C. L. Zitnick, and D. Parikh. "Cider: Consensus-based image description evaluation". In: *CVPR*. 2015, pages 4566–4575.

PRIDE-2: Improving Personal Risk Detection Based on One-Class Classifiers and the Use of Wearable Sensors

Jorge Rodríguez[1], Miguel Angel Medina-Pérez[1], Luis A. Trejo[1],
Ari Y. Barrera-Animas[1], Raúl Monroy[1], Armando López-Cuevas[1], and
José Ramírez-Márquez[2]

[1] Tecnologico de Monterrey, México
{jorger,migue,ltrejo,A01373306,raulm,acuevas}@itesm.mx
[2] Stevens Institute of Technology, Hoboken, NJ, USA
jmarquez@stevens.edu

In our previous project [2], we defined personal risk detection as the timely identification of a situation when someone is at imminent peril, such as a health crisis or a car accident. A risk-prone situation should produce sudden and significant deviations in user patterns, and the changes can be captured by a group of sensors, such as an accelerometer, gyroscope, and heart rate monitor, which are normally found in current wearable devices. Previous research findings were published in [2, 11] and presented at HPI Future SOC Lab. The present work rises with the aim of improving our previous results. In order to achieve it, the following three approaches were tested: 1) a visualization method in real-time of PRIDE users leveraged with a one-class classifier called Bagging-TPMiner, 2) the addition of frequency-domain features to the time-domain features embraced in the PRIDE dataset, and 3) improve the accuracy obtained by previous one-class classifiers through testing a cluster validation algorithm. We were able to report part of our results in [8], which have been recently submitted for publication. Although experiment results reported in this document are encouraging, due to the sheer amount of data, the results presented in this report are partial. In order to fulfil the experiments, we are submitting an extension at HPI Future SOC Lab for the period that ends on April 2017.

1 Introduction

The work presented is a natural continuation of previous work performed using a 64-core cluster with 128 GB RAM from HPI Future SOC Lab. Thanks to this support we have been able to report our findings in [2, 11] as mentioned in our technical report for the period that ended on November 2016 and that was presented at HPI Future SOC Lab Day - Fall 2016.

In our previous project, we defined personal risk detection as the timely identification of a situation when someone is at imminent peril, such as a health crisis or a car accident. We worked under the hypothesis that a risk condition produces sudden and significant deviations regarding standard physiological and behavioral user patterns. Monitoring for the occurrence of these changes can be done using a group of sensors, such as the accelerometer, gyroscope, heart rate, etc. Recently we released a dataset, called PRIDE [2] that provides a baseline for the development and the fair comparison of personal risk detection mechanisms.

In this stage of our research, our intention is to improve our previous results. Three approaches were chosen to accomplish our objective. The first one is a novel visualization method, currently under revision [8], to track and identify in real-time when a person is in a risk-prone situation. The visualization is leveraged with a traffic light model of one-class classifiers called Bagging-TPMiner, introduced in [10]. Bagging-TPMiner was tested and compared with the classifiers tested in previous work presented at HPI Future SOC Lab Day - Fall 2016. The second approach is the generation of features in the frequency-domain combined with ones in the time-domain, reported in [2]. Our hypothesis here is that an increase in accuracy can be achieved using both types of features. Since our previous results relied on clusters of different projections of the data [2, 11], the third approach we are undertaking is obtaining the best value for the number of clusters. To do this, we are currently developing and testing a cluster validation algorithm.

2 Datasets and methods

We are relying on two different repositories of datasets. The first is the personal risk detection (PRIDE) dataset repository, which contains 23 datasets, each one comprised of the records obtained from observing the health measurements of different users, with diverse characteristics regarding gender, age, height, and lifestyle. The second is the University of California, Irvine(UCI) repository, which contains various datasets with numerical and nominal features, generally used for supervised classification.

2.1 The personal risk detection (PRIDE) dataset repository

PRIDE test subjects are comprised of eight female and 15 male volunteers, aged between 21 and 52 years, with heights between 1.56 m and 1.86 m, and weights between 42 kg to 101 kg. The volunteers exercising rates ranged from 0 to 10 hours a week, and the time they spent sitting ranged from 20 to 84 hours a week. The health measurements were done using the sensors on the Microsoft Band v1©, recording the values of the sensors using a mobile application developed using the available SDK, and installed in each user's smartphone. The used sensors from the band and the frequencies of data acquisition for that sensor are described in Table 1.

The data collected from the activities of the user in one week comprises the Normal Conditions Data Set (NCDS), which can be used to construct the normal behavior baseline, which will be used to look for deviations in the behavior, and thus detect risk situations. The same 23 test subjects participated in another data acquisition process to test how the users responded when confronted with an anomalous situation. They needed to perform the following activities: rushing 100 meters as fast as possible, going up and down the stairs in a multi-floor building as quickly as possible, a two-

Table 1: Description of the Microsoft Band Sensors

Sensor	Description	Frequency
Accelerometer	Provides X, Y, and Z acceleration in g units. 1 g = 9.81 meters per second squared (m/s^2).	8 Hz
Gyroscope	Provides X, Y, and Z angular velocity in degrees per second, (°/s) units.	8 Hz
Distance	Provides the total distance in centimetres, current speed in centimetres per second (cm/s), current pace in milliseconds per meter (ms/m).	1 Hz
Heart Rate	Provides the number of beats per minute, also indicates if the heart rate sensor is fully locked onto the wearer's heart rate	1 Hz
Pedometer	Provides the total number of steps the user has taken.	1 Hz
Skin Temperature	Provides the current skin temperature of the user in degrees Celsius.	33 mHz
UV	Provides the current ultraviolet radiation exposure intensity (None, Low, Medium, High, Very High)	16 mHz
Calories	Provides the total number of calories burned by the user.	1 Hz

Table 2: PRIDE feature vector structure (1–18 fields)

Gyroscope Accelerometer						Gyroscope Angular Velocity						Accelerometer					
X axis		Y axis		Z axis		X axis		Y axis		Z axis		X axis		Y axis		Z axis	
$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

minute box practice session, falling back and forth, and holding one's breath for as long as possible. Each activity aims to simulate a dangerous or abnormal situation in the real world, e.g., running away from a dangerous situation, evacuating a building during an emergency, defending from an aggressor, swooning, and experiencing breathing problems such as dyspnea. The records of these scenarios comprise the Anomalous Conditions Data Set (ACDS).

2.2 Visualization of personal risk-prone situations using PRIDE dataset

The PRIDE dataset was preprocessed according to [2], to perform any experiments with one-class classifiers. Tables 2 and 3 show the structure of the 26-dimensional feature vector of each object after the preprocessing phase, where $\bar{\mathbf{x}}$, $\mathbf{s}$, and Δ stands for mean, standard deviation, and delta of the value, respectively.

2.2.1 Classifiers tested over the PRIDE dataset

The visualization model is capable of providing a decision maker a visual description of the physiological behaviour of an individual, or a group thereof; through it, the decision maker may infer whether further assistance is required or if a risky situation is in progress. The visualization is leveraged with a traffic light model of a one-class

Table 3: PRIDE feature vector structure (19–26 fields)

Heart Rate	Skin Temperature	Pace	Speed	UV	Δ Pedometer	Δ Distance	Δ Calories
19	20	21	22	23	24	25	26

classifier. This combination allows us to train the decision maker into visualizing correct and potential risky or abnormal behaviour. Tested one-class classifiers were the following:

- **ocSVM**: The implementation of ocSVM [13] included in LibSVM [3] with the default parameter values ($\gamma = 0.038$ and $\nu = 0.5$) and using the radial basis function kernel.
- **Parzen**: Parzen window classifier using the Euclidean distance [5]. For every training dataset, the classifier computes the width of the Parzen-window by averaging the distances between objects sampled every 60 s.[3]
- **k-means1**: A version of the Parzen window classifier based on k-means [12]. k-means1 classifies new objects based only on the closest centre of the cluster.
- **k-means2**: A version of the Parzen window classifier based on k-means [7]. k-means2 classifies new objects using all the centres of the clusters.
- **OCKRA**: A ensemble of one-class classifiers based on k-Means++ [1] for PRIDE [2]. The ensemble is basically a combination of 100 one-class classifiers based on k-Means++ clustering.
- **Bagging-TPMiner**: An ensemble of one-class classifiers introduced in [10]. Unlike OCKRA, Bagging-TPMiner builds individual classifiers based on different subsets of objects and using all the features.

2.3 Preprocessing PRIDE in time-/frequency-domain for online personal risk detection

In this approach, the PRIDE dataset was used in its raw form in order to calculate the frequency-domain features and add them to the time-domain features obtained by following the procedure proposed in [2]. The time and frequency-domain features were only calculated for the 3-axis gyroscope and accelerometer sensors. Heart rate, Skin temperature, Delta pedometer, Delta distance, Speed, Pace, Delta calories, and UV features remain as detailed in Table 3. Accordantly, feature vector represents the sensors measurements in an interval of one second.

[3] This procedure saved approximately 7 days when computing the distances per test subject using an Intel Core i7-4600M CPU at 2.90 GHz.

2.4 Clustering Validation

One way to determine the best clusters for any dataset is using experts that determine the belonging of one object to a group. However, this constraint is not realistic by the sheer number of information. Our hypothesis for the third approach is that experts can be simulated with classifiers to determine if a partition done by a clustering algorithm is correct or not. Thus, we are constructing a CVI that relies on an ensemble of different classifiers, which evaluate the partition using the cluster assignment as class labels and observing the capabilities of the ensemble to correctly discern the belonging cluster/class of a new object. A correct clustering would allow the ensemble to distinguish between the different groups properly.

Our CVI takes the output of a clustering algorithm, where each object has its corresponding cluster label. The input data is separated in 5 folds, each taking 80 % for training and 20 % for testing. The ensemble, composed of 1-nearest neighbor, multilayer perceptron, support vector machine, Bayesian Network, and Random Forest, trains using each fold training dataset and each classifier predicts the class for each object in the testing dataset. The class of an object is selected by majority voting. Lastly, the Area Under the Curve is computed for each fold and averaged.

3 Experiments

All experiments were performed using a 64-core cluster with 128 GB RAM from HPI Lab and 32 cores from ITESM CEM. Approximately, a total of 1 366 200 hours (56 925 days) were required to perform the visualization experiments; where CPU hours = 10 days (time to train and test the one-class classifiers) × 49 values of k in k-means algorithm × 5 folds × 23 users + 25 days (time to evaluate the one-class classifiers) × 23 users. Calculations to transform the PRIDE dataset to time and frequency-domain approximately took 184 hours (8 days); where CPU hours = 7 minutes (to obtain time and frequency-domain features) × 1 user day log × 7 user's day logs × 23 users. For the CVI experiments, evaluating a dataset consists of 5 folds × 5 classifiers × 99 values of k × 5 clustering algorithms, resulting in 12 376 classifiers being constructed. We are using the infrastructure provided by HPI to evaluate the biggest 50 datasets from a repository of 80.

3.1 Visualization over PRIDE

Following our previous work [11], we tested the performance of all the classifiers using *five-fold cross-validation* for all the 23 users of the PRIDE repository. For each classifier, we computed the Area Under the Curve (AUC) of the true positive (true risk-prone situation) detection rate (TP) versus the false positive (false risk-prone situation) detection rate (FP). This indicator gives an idea of the general accuracy of the classifier for all false positive detection rates.

To study differences between the algorithms' performance, we use the Friedman's test [4] and the Bergmann-Hommel's dynamic post-hoc [6], with a level of signif-

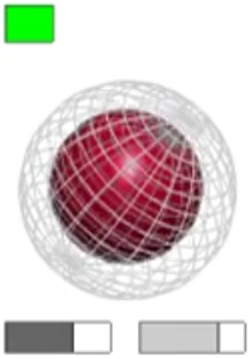


Figure 1: FiToViz model: visualization scheme of concentric spheres encoding sensor data

icance $\alpha = 0.05$. To show these results, we use *Critical Difference (CD) diagrams* [4], because they succinctly present the rank of an algorithm for an accuracy indicator. CD diagrams also show both the magnitude and the significance of any difference between the algorithms' performance [4]. Note that, in particular, the best algorithm appears rightmost, and statistically similar algorithms appear joined with a thick line.

To support real-time decision making through the use of wearables, we have designed a novel visualization approach proposed in [8]. The visualization consists on mapping the values provided by the sensors of the Microsoft Band v1© into a dynamic image that is both, informative and visually intuitive to the human eye.

The core of the visualization model is formed by two concentric spheres as depicted in Figure 1, where every variable describing the size and position of the spheres is matched to a sensor value. In this sense, the spheres are dynamic in accordance with the variations of the data. In total, the visualization model displays 11 variables. Of these, the inner sphere represents changes of six variables, namely: UV exposure, heart rate, skin temperature, and 3-axis accelerometer. The outer sphere encodes three variables of the 3-axis gyroscope, and two more variables, distance, and calories are encoded as bar graphs.

3.2 PRIDE in the frequency-domain

Frequency-domain features are calculated using the discrete fast Fourier transform [9]. Table 4 shows the frequency and time-domain features calculated for the 3-axis gyroscope accelerometer, gyroscope angular velocity, and accelerometer measurements. Currently, we have 102 time-domain features, 90 frequency-domain features, and eight non-motion features. Thus, a 200-dimensional feature vector represents each one-second observation.

Table 4: Features from time and frequency domain of accelerometer and gyroscope sensors

Time-domain	Frequency-domain
Signal magnitude area *	FFT Energy
Root mean square	FFT Mean Energy
Signal Vector Magnitude *	FFT Std Dev Energy
Average Signal Vector Magnitude *	Peak Power
Variance sum *	Peak DFT Bin
Curve length	Peak Magnitude
Average non linear energy	Entropy
Variance	Spectral Entropy
Mean	Peak Frequency
Max	Peak energy
Min	
Standard Deviation	
Median	
Range	

* features calculated with the 3 axes of each sensor

3.3 Testing the CVI

To determine the best k and clustering algorithm, we used K-means, Expectation Maximization, Single Linkage, Lineal Vector Quantization and Self-Organizing Maps as the clustering algorithms to test. Each algorithm is used on each dataset with a value of k from 2 to 100, to have different possible groups. Then, the best k is selected as the partition that has the highest value using our CVI. We are also testing our CVI against Dunn and Davies-Bouldin, which are acclaimed CVI in the literature. This process is done for 80 datasets from the UCI repository.

4 Results and discussion

The experiments done for visualization show a better manner to determine whenever a person is in risk, allowing a decision maker to make an informed decision based on the visual aids. Our CVI results are preliminary, but the results are encouraging that it can be a better index than the one proposed on the literature. For the conversion of the features in PRIDE to the frequency-domain, we are still working on this process so our results are only partial. In this section we explore with further detail the results of all these experiments.

Table 5: Area (percentage) under the curve for True Positive Rate versus False Positive Rate

Test Subject	Bagging-TPMiner	ocSVM	Parzen	k-means1	k-means2	OCKRA
Average	87.5	86.4	88.6	88.5	86.8	89.1

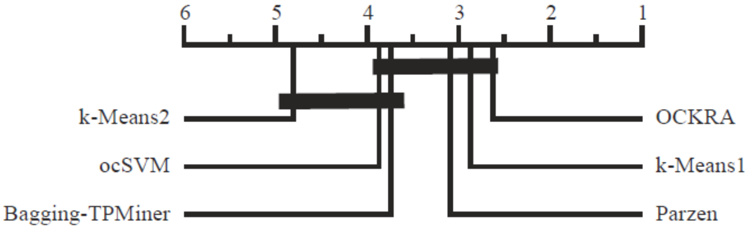


Figure 2: Statistical comparisons of 6 one-class classifiers evaluated on the 23 users of PRIDE

4.1 Visualization over PRIDE

Table 5 shows that, in average, OCKRA is still the best classifier. Moreover, Bagging-TPMiner performs worse than Parzen and k-Means1.

According to the Friedman's test, there is a significant statistical difference among the algorithms. Figure 2 shows that, according to the Bergmann-Hommel's dynamic post-hoc, there is not a significant statistical difference among OCKRA, k-Means1, Parzen, Bagging-TPMiner, and ocSVM. Nevertheless, OCKRA achieves the best average ranking. Hence, we propose using OCKRA for complementing the visual model.

Regarding the visualization, two videos that show one user activity during a period can be found at the following link: https://youtu.be/07G8HNXvIEc and https://youtu.be/m0LZHDTKk5E. Every minute in the video represents one hour of user activity. Extended versions of the video are also available upon request to the authors. Experiments results show that there are several aspects in the visualization that can be naturally assimilated by the decision maker through continuous observation of the visualization; for example, the dynamic changing radius of the inner sphere simulates a heart beating. In Figure 3, four different activities performed by an individual are shown: a) Normal activity; Simulation of a risk situation: b) evacuation alert, c) running away, and d) fighting back. It is important to notice that normal activity differs highly from risk activities.

4.2 PRIDE in the time-/frequency-domain

Transforming the PRIDE dataset features to time and frequency-domain is a fist step that will enable us to perform experiments with one-class classifiers. We believe that both types of features will allow us to obtain a better representation of quotidian standard physiological and behavioral user patterns. In consequence, we

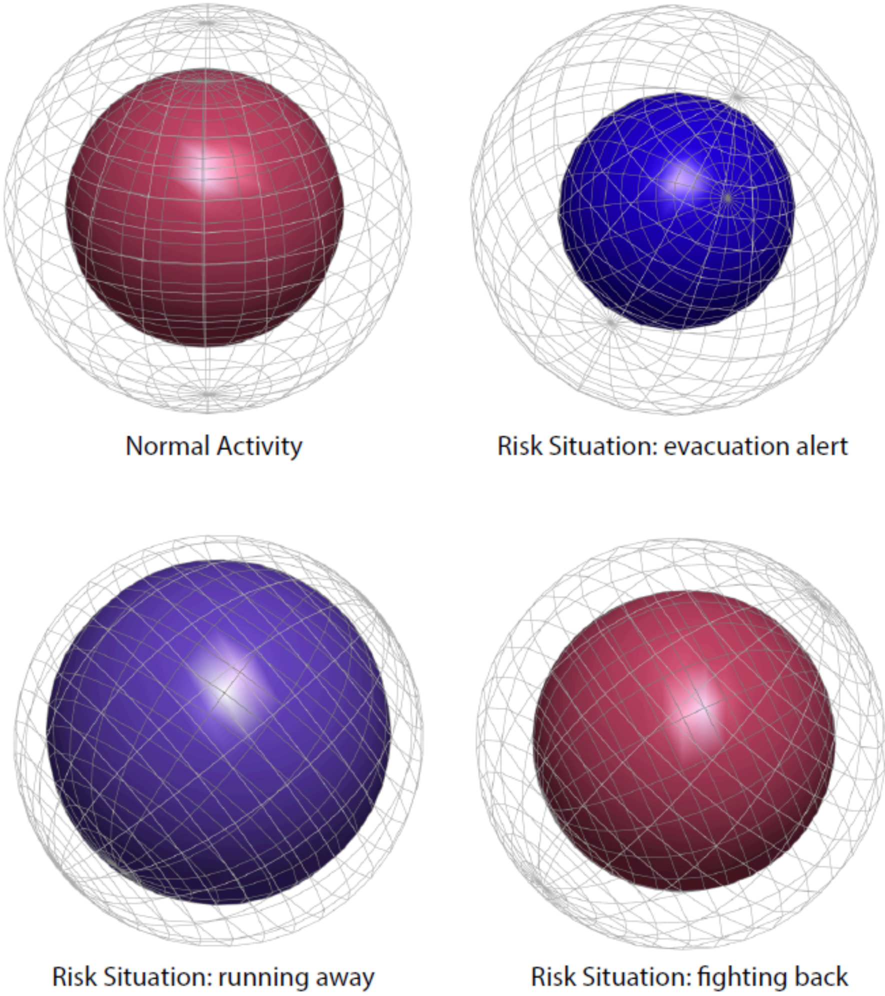

Normal Activity
Risk Situation: evacuation alert
Risk Situation: running away
Risk Situation: fighting back

Figure 3: Four different types of activities visualized: a) Normal activity; Simulation of a risk situation: b) evacuation alert, c) running away, and d) fighting back

Table 6: Number of times a CVI found the number of k/Number of times a CVI found the nearest number of k

Clustering Algorithm	Dunn	Davies-Bouldin	Ensemble CVI
EM	5/5	5/8	***10**/**21***
KMeans	**11**/12	6/6	*8/**24***
LVQ	**21**/5	20/6	*20/**12***
SingleLinkage	16/9	13/13	***23**/**16***

can increase the accuracy of the classification task of one-class classifiers proposed in [11]. However, we are anticipating an incremental amount of CPU hours due to moving from 26-dimensional feature vectors to 200-dimensional feature vectors, for all 23 users.

4.3 Preliminary results of the CVI

Currently, we have finished the clustering and classification step on 50 of the 80 datasets. Our preliminary experiment follows the methodology done in the area of cluster validation, where the number of k obtained for a clustering algorithm on a dataset is compared to the number of classes that a dataset for supervised classification has. For our results in Table 6, we record how many times using each CVI the correct number of k can be found. If no CVI finds the exact number, we mark who was the closest one to the number of classes in a dataset. In asterisks, we show the CVI that obtained a closer number to the real number of classes. Preliminary results show that our CVI has the potential of finding the correct class more times than the other ones.

5 Conclusions and further work

Results obtained from the three approaches previously presented are encouraging, by getting us a step closer to our aim of improving previous results presented at HPI Future SOC Lab Day – Fall 2016. We were able to report part of our results in [8], which have been recently submitted for publication. Also, the results of our CVI using an ensemble of classifiers are encouraging and will allow us to improve the accuracy to detect risk prone detection and propose a better method for determining the number of groups for clustering problems. However, due to the sheer amount of data used for all the experiments, the results presented in this report are partial. To complete the experiments with all the data and provide thoroughly tested algorithms and results, we are submitting an extension on the use of the 64 cores VM with 128 GB RAM from the HPI Future SOC Lab period that ends on April 2017.

References

[1] D. Arthur and S. Vassilvitskii. "k-means++: The Advantages of Careful Seeding". In: *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2007, pages 1027–1035.

[2] A. Y. Barrera-Animas, L. A. Trejo, M. A. Medina-Pérez, R. Monroy, J. B. Camiña, and F. Godínez. "Online Personal Risk Detection Based on Behavioural and Physiological Patterns". In: *Information Sciences* 384 (Apr. 2017), pages 281–297. DOI: doi:10.1016/j.ins.2016.08.006.

[3] C.-C. Chang and C.-J. Lin. "LIBSVM: A Library for Support Vector Machines". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 2.3 (2011), pages 1–27. DOI: 10.1145/1961189.1961199.

[4] J. Demšar. "Statistical Comparisons of Classifiers over Multiple Data Sets". In: *Journal of Machine Learning Research* 7 (Dec. 2006), pages 1–30.

[5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. 2nd. Wiley-Interscience, 2001. 680 pages.

[6] S. García and F. Herrera. "An Extension on Statistical Comparisons of Classifiers over Multiple Data Sets for all pairwise comparisons". In: *Journal of Machine Learning Research* 9.2677-2694 (2008), page 66.

[7] G. Giacinto, R. Perdisci, M. D. Rio, and F. Roli. "Intrusion Detection in Computer Networks by a Modular Ensemble of One-Class Classifiers". In: *Information Fusion* 9 (1 Jan. 2008). Special Issue on Applications of Ensemble Methods, pages 69–82. DOI: 10.1016/j.inffus.2006.10.002.

[8] A. López-Cuevas, M. A. Medina-Pérez, R. Monroy, J. Ramírez-Márquez, and L. A. Trejo. "FiToViz: A Visualisation Approach for Real-time Risk Situation Awareness - (under revision)". In: *IEEE* (2017).

[9] A. Maxhuni, A. Muñoz-Meléndez, V. Osmani, H. Perez, O. Mayora, and E. F. Morales. "Classification of bipolar disorder episodes based on analysis of voice and motor activity of patients". In: *Pervasive and Mobile Computing* 31 (2016), pages 50–66. DOI: 10.1016/j.pmcj.2016.01.008.

[10] M. A. Medina-Pérez, R. Monroy, J. B. Camiña, and M. García-Borroto. "Bagging-TPMiner: a classifier ensemble for masquerader detection based on typical objects". In: *Soft Computing* 21.3 (2017), pages 557–569. ISSN: 1433-7479. DOI: 10.1007/s00500-016-2278-8.

[11] J. Rodríguez, A. Y. Barrera-Animas, L. A. Trejo, M. A. Medina-Pérez, and R. Monroy. "Ensemble of One-Class Classifiers for Personal Risk Detection Based on Wearable Sensor Data". In: *Sensors* 16.10 (2016), page 1619. DOI: 10.3390/s16101619.

[12] D. M. J. Tax and R. P. W. Duin. “Combining One-Class Classifiers. Second International Workshop, MCS 2001”. In: *Multiple Classifier Systems*. Volume 2096. Springer Berlin Heidelberg, July 2001, pages 299–308. DOI: 10.1007/3-540-48219-9_30.

[13] V. N. Vapnik. *Statistical Learning Theory*. 1st. Volume 1. Wiley-Interscience, 1998. 768 pages.

Application of Reverse Time Migration to Ultrasonic Echo Data in Non-destructive Testing

Maria Grohmann and Ernst Niederleithinger

Federal Institute of Materials Research and Testing, BAM, Berlin
{firsname.lastname}@bam.de

Ultrasonic echo testing is widely used in civil engineering as a non-destructive testing method for imaging and investigation of concrete structures; in particular to measure thickness and to locate and characterize built-in components or inhomogeneities. Mostly used for imaging are synthetic aperture focusing techniques. These algorithms are highly developed but have some limitations. For example, it is not possible to image the lower boundary of built-in components such as tendon ducts or vertical reflectors.

To improve the imaging of concrete structures, we adapted a geophysical imaging technique, reverse time migration, for non-destructive testing. By using the entire wave field, including waves reflected more than once, there are fewer limitations compared to synthetic aperture focusing techniques. As a drawback, the required computation cost is significantly higher as for the techniques currently used.

Simulations for a concrete structure and following experiments at a concrete specimen demonstrate the potential of our technique for non-destructive testing.

Vertical reflectors inside the specimen were imaged clearly using reverse time migration. Such structures cannot be imaged by conventional techniques. Hence, reverse time migration advances ultrasonic testing in civil engineering.

1 Introduction

The ultrasonic echo technique is an important test method used in non-destructive testing (NDT) to determine the interior structure of concrete building elements [6, 10]. Important NDT tasks include thickness measurements, the localization of cracks and debonding as well as the localization and characterization of built-in components and inhomogeneities. The available Synthetic Aperture Focusing Techniques (SAFT) [5, 7, 11] for the reconstruction of ultrasonic echo data have difficulties in imaging vertically dipping interfaces as shown in Figure 1. The SAFT image of ultrasonic measurements at a foundation slab shows the successful reconstruction of the backwall of the slab. However vertical reflectors inside the slab (marked with red ellipses) could not be imaged.

Compared to SAFT, the geophysical imaging method reverse-time migration (RTM) has the potential to produce a more complete imaging of the features inside the concrete specimens. Thus RTM was applied in this work to experimental ultrasonic echo

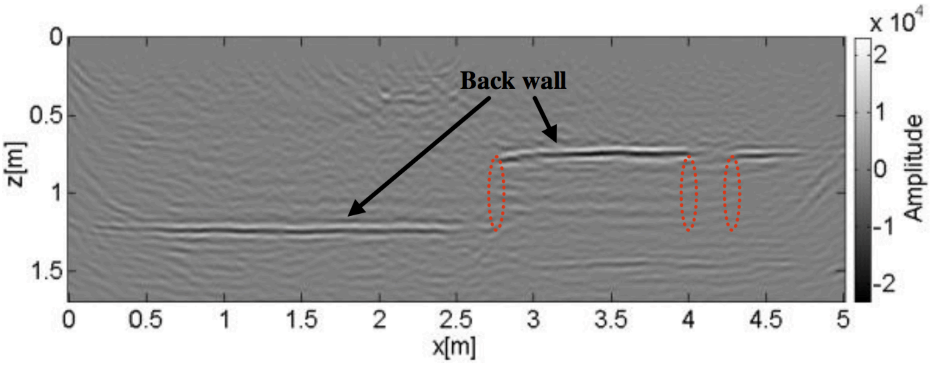


Figure 1: SAFT result of measurements at a foundation slab [4]

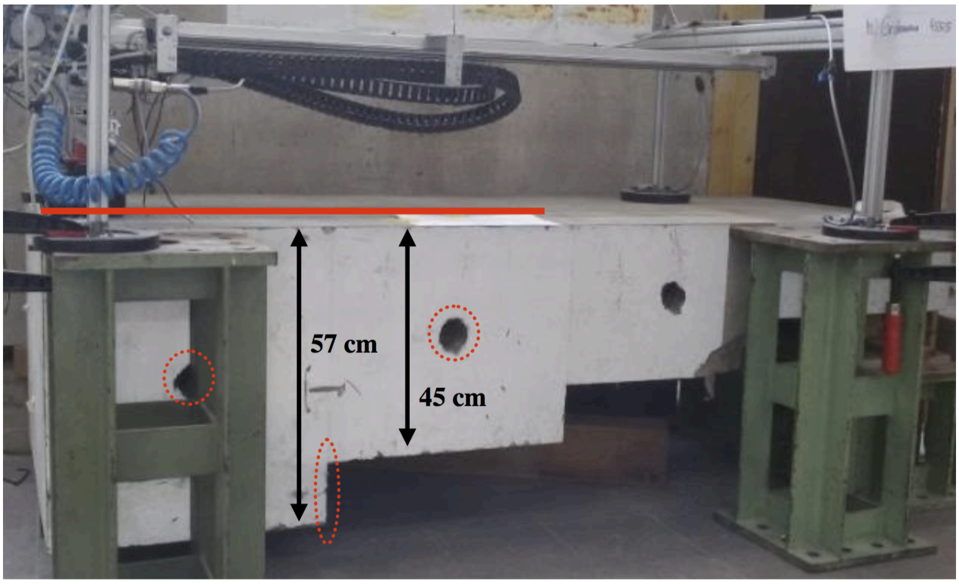


Figure 2: Concrete specimen consisting of several steps and empty tendon ducts.

data acquired at a concrete specimen consisting of several steps and empty tendon ducts.

2 Ultrasonic Measurements at a Concrete Specimen

Reverse-time migration was tested with shear wave (sh) data that were recorded on a concrete specimen incorporating several steps at the back and empty tendon ducts. Figure 2 shows the test specimen as well as the line on which the measurements were taken (marked in red). Our objectives were the reconstruction of the vertical step and the determination of the shape of the tendon ducts (marked with red ellipses in Figure 2).

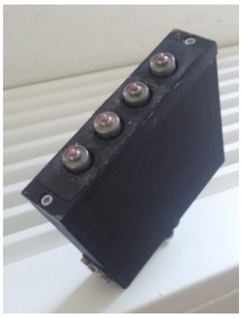


Figure 3: Shear wave transducer consisting of four point contact transducers.

Table 1: Measurement parameters

Parameter	
Number of Source positions	44 (distance 2 cm)
Number of Receiver positions	86 (distance 1 cm)
Measurement frequency [kHz]	50 kHz
Recording Time [s]	0.0025 s

We used a multistatic arrangement to collect the ultrasonic echo data. Two ultrasonic transducers (one for transmitting and one for receiving ultrasonic waves) were moved over the surface. Both were separated from each other and changed their positions and distances. Figure 3 shows one of the two shear wave transducers used. Four point contact transducers are connected in parallel and mounted in a single casing. Using a scanner system, the transmitter and receiver transducer are moved automatically from one measurement point to another. Table 1 shows the measurement parameters.

3 Reverse Time Migration

RTM is a wave-equation based imaging algorithm. It is a standard imaging technique in the seismic industry and was introduced by McMechan [8] and Baysal et al. [1]. Müller et al. [9] proved the applicability of RTM in NDT to image synthetic ultrasonic data generated with polyamide- and concrete-like models.

RTM is a wave field-continuation method in time and uses the full wave equation. It is able to image reflectors even with steep dips and strong velocity contrasts. A major disadvantage is the required extensive computation and memory capacity. The RTM implementation in this work uses numerical solutions of the 2D elastic wave equation. The algorithm we used is based on a two dimensional finite difference modeling code created by using the Madagascar software package [2]. In addition to the 2D elastic code, which takes into account all wave types (shear waves, longitudinal waves and Rayleigh waves), we implemented a 2D elastic SH-RTM algorithm (SH: horizontally

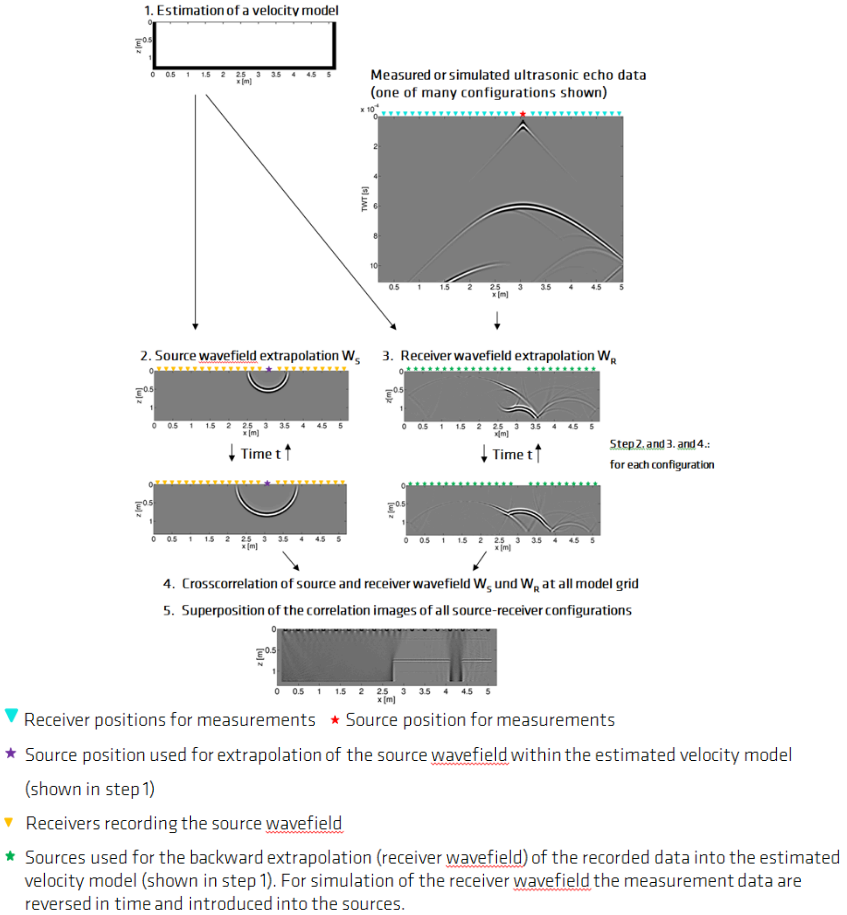

1. Estimation of a velocity model
Measured or simulated ultrasonic echo data
(one of many configurations shown)
TWT [s]
x [m]
z [m]
2. Source wavefield extrapolation W_S
3. Receiver wavefield extrapolation W_R
Time t
Time t
Step 2. and 3. and 4.:
for each configuration
4. Crosscorrelation of source and receiver wavefield W_S und W_R at all model grid
5. Superposition of the correlation images of all source-receiver configurations
▼ Receiver positions for measurements
★ Source position for measurements
★ Source position used for extrapolation of the source wavefield within the estimated velocity model (shown in step 1)
▼ Receivers recording the source wavefield
★ Sources used for the backward extrapolation (receiver wavefield) of the recorded data into the estimated velocity model (shown in step 1). For simulation of the receiver wavefield the measurement data are reversed in time and introduced into the sources.

Figure 4: Principle of Reverse Time Migration [3]

polarized shear waves). The latter is of particular importance since SH-waves, which do not convert to other types of waves at interfaces, are typically used for ultrasonic echo measurements.

RTM consists of the following main steps (Figure 4):

1. Estimation of a velocity model.
2. The source wave field W_S is extrapolated forward in time using the source location, the source wavelet and the estimated velocity model (from 1.). The scattered wave field is recorded at the receiver positions.
3. The receiver wave field W_R is propagated backward in time, from all receiver locations using the recorded data (in our case the recorded ultrasonic data) as boundary condition as well as the estimated velocity model (from 1.).
4. The imaging condition used here computes the zero lag of the local cross-correlation between the two simulation results at all model grid points to find positions of existing subsurface reflectors.
5. For the final result, the correlation images of all configurations are stacked.

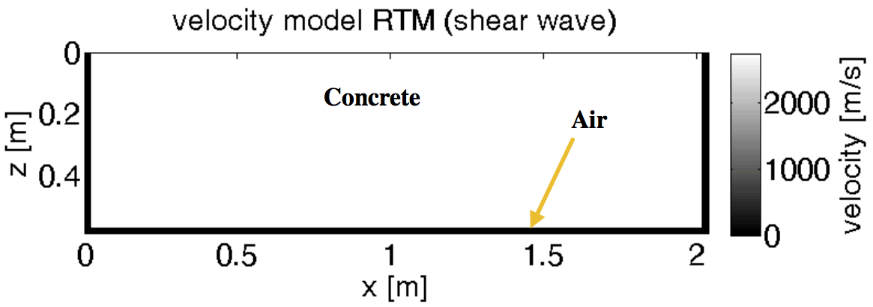


Figure 5: Velocity model used for RTM

Table 2: Material parameters

Parameter	
Velocity longitudinal wave [m/s]	Concrete: 4400 / Air: 333
Velocity shear wave [m/s]	Concrete: 2750 / Air: 0
Density [g/cm^3]	Concrete: 2.4 / Air: 0.0012

4 Reverse Time Migration Results

The velocity model we used for RTM (see step 1 in section 3) is shown in Figure 5 and the corresponding material parameters are listed in Table 2. The outer limits of the concrete specimen were assumed to be known.

The migration parameters are listed in Table 3. For one RTM result 44 measurement data sets had to be evaluated. One data set comprises 86 time signals (traces) with a length of 80 000 samples.

We used 22 nodes of the HPI FUTURE SOC LAB 1000 Core Cluster for calculating a part of our RTM results. We had access to the cluster for a total of three weeks (one week per month from January 2017 to March 2017), which enabled us to analyze 22 data sets using RTM. Figure 6 shows an example of a data set processed with a

Table 3: Migration parameters

Parameter	
Model size	4080 × 1180 grid points
Distance between grid points [mm]	0.5 mm
Frequenz Ricker Wavelet [kHz]	50 kHz
Time step [s]	0.000 000 01 s
Simulation time [s]	0.0008 s
Number of sources	44 (distance 2 cm)
Number of receivers	86 (distance 1 cm)

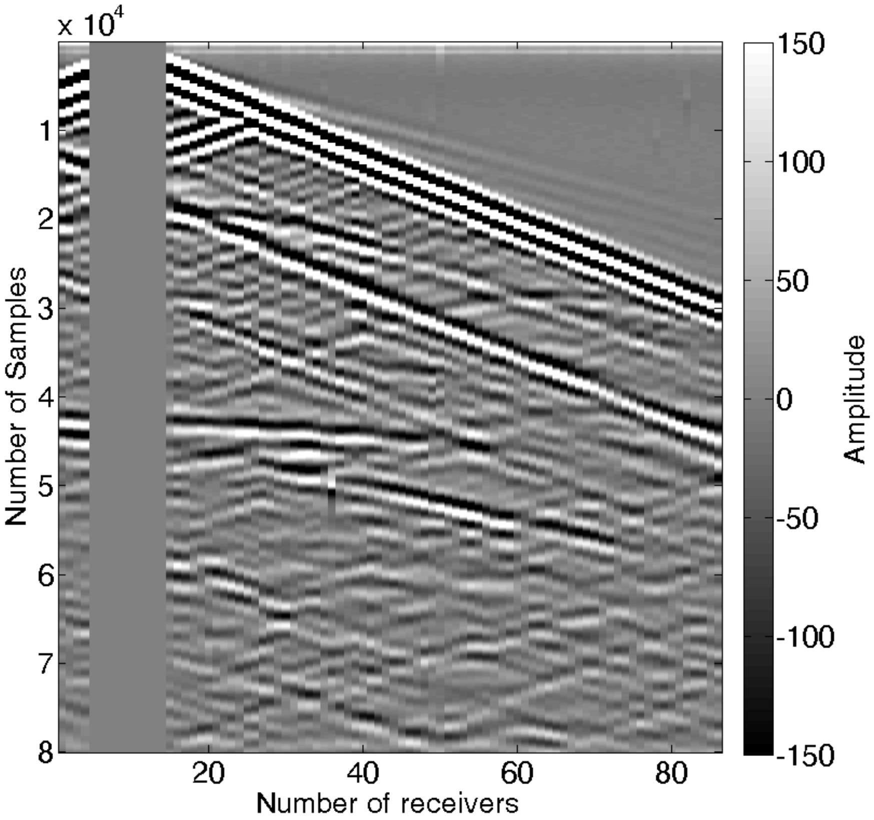


Figure 6: Measured ultrasonic data

bandpass filter. The measured amplitudes are color-coded. There are ten traces in the data set (receiver position 5 to 14) where no data could be obtained because of the dimensions of source and receiver transducer.

For a first evaluation, the measurement data were processed with a bandpass filter (cut-off frequencies: 8 kHz/150 kHz) before applying RTM. Furthermore an automatic gain control (AGC) as well as a 3D/2D correction was applied to the measurement data since the RTM algorithm is a 2D implementation and our data originate from a 3D source and a 3D medium. The corresponding RTM results are shown in Figure 7 and 8.

Both RTM results show a clear and sharp reconstruction of the vertical edge of the step (marked red in Figure 7) as well as the imaging of the top boundaries of the two tendon ducts (marked with yellow arrows in Figure 7). The lower boundaries of both tendon ducts could not be reproduced. This may be due to an inaccurate migration velocity or inadequate simulation time. Therefore, migrations with different velocity values as well as simulation times are currently in progress.

For the result in Figure 10, the first step of the specimen was integrated into the velocity model (Figure 9) since it is known from the first RTM reconstruction results. The second step is now clearly visible (marked red in Figure 10). This result

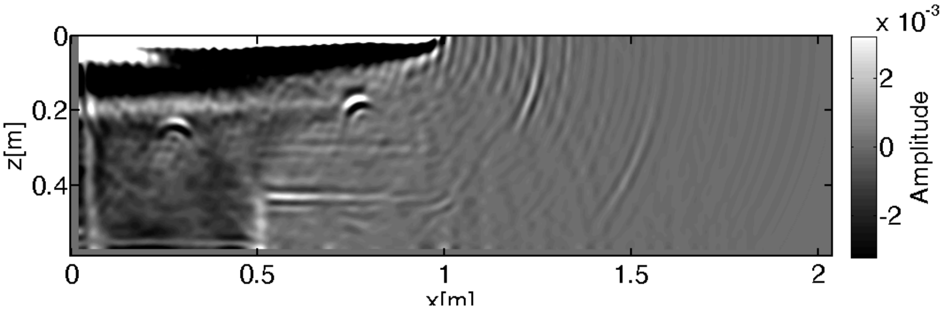


Figure 7: RTM result after applying a bandpass filter to the measurement data.

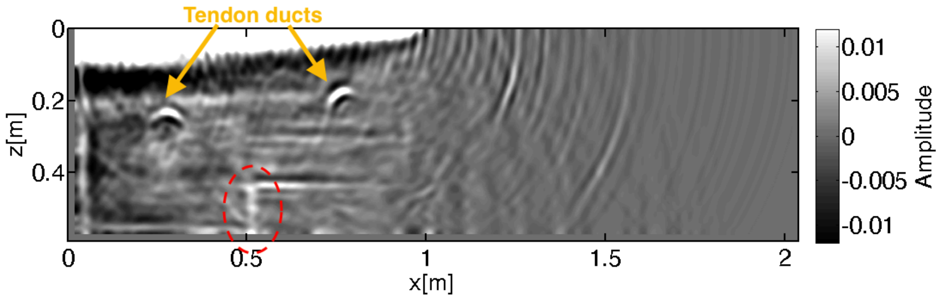


Figure 8: RTM result after applying a bandpass filter, AGC and 3D/2D correction to the measurement data.

demonstrates that for complex models a multistage RTM is required if the a priori information about the structure is insufficient.

In summary the presented RTM results clearly illustrate the advantages of RTM since vertical reflectors inside the specimen could be imaged. The reconstruction of the structure of the lower boundary of the investigated concrete specimen could be improved by using RTM.

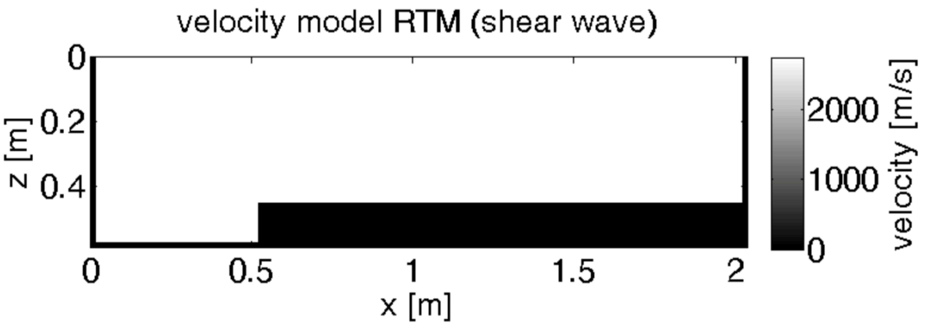


Figure 9: Velocity model used for RTM

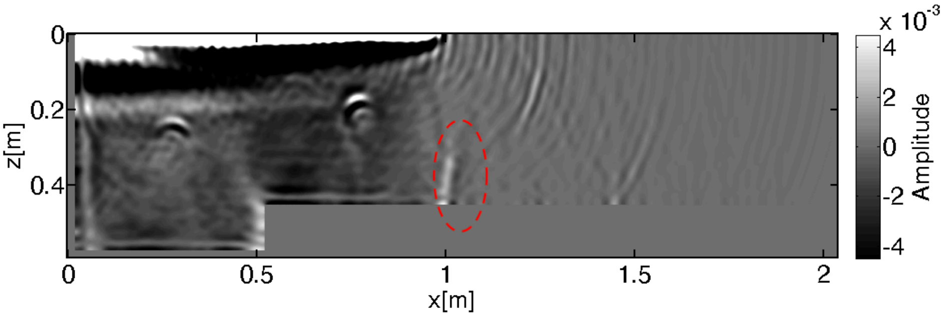


Figure 10: RTM result after applying a bandpass filter to the measurement data and using the velocity model shown in Figure 9.

5 Outlook

An interesting direction for future research is to evaluate the potential of the 2D RTM code for the analysis of data recorded with longitudinal transducers.

To gain a more precise comparison between RTM and SAFT the collected data should be also evaluated by using the 2D SAFT algorithm.

In addition to the measurements on the concrete specimen presented within this report further measurements on a polyamide specimen containing a borehole are planned. Our objective is the imaging of the full shape of the borehole using elastic RTM.

Furthermore RTM artefacts have to be analyzed and eliminated. For this task, alternatives to the cross-correlation imaging condition as well as pre-imaging filtering techniques may be used. In addition, the algorithm should be expanded to 3D and the full elastic wave equation. The use of adopted source signals should improve the image quality as well.

Another topic to be addressed is how to account for the size of the ultrasonic arrays. The RTM codes used in this work assume point sources. Hence, the migration algorithm does not calculate the source and receiver wave fields fully correctly.

References

[1] E. Baysal, D. D. Kosloff, and J. W. C. Sherwood. "Reverse time migration". In: *Geophysics* 48 (Nov. 1983), pages 1514–1524. DOI: 10.1190/1.1441434.

[2] S. Fomel, P. Sava, I. Vlad, Y. Liu, and B. V. "Madagascar: open-source software project for multidimensional data analysis and reproducible computational experiments". In: *Journal of Open Research Software* (2013).

[3] M. Grohmann, S. Müller, E. Niederleithinger, and S. Sieber. "Reverse Time Migration: Introducing a New Imaging Technique for Ultrasonic Measurements in Civil Engineering". In: *Near Surface Geophysics* (2017). in press.

[4] M. Grohmann, E. Niederleithinger, and S. Buske. "Geometry determination of a foundation slab using the ultrasonic echo technique and geophysical migration methods". In: *Journal of Nondestructive Evaluation* 1 (2016). DOI: 10.1007/s10921-016-0334-z.

[5] M. Krause, B. Milmann, F. Mielentz, D. Streicher, B. Redmer, K. Mayer, K.-J. Langenberg, and M. Schickert. "Ultrasonic Imaging Methods for Investigation of Post-Tensioned Concrete Structures: A Study of Interfaces at Artificial Grouting Faults and its Verification". In: *Journal of Nondestructive Evaluation* 27.1-3 (2008), pages 67–82. DOI: 10.1007/s10921-008-0033-5.

[6] S. M. "Progress in ultrasonic imaging of concrete". In: *Materials and Structures* 38.11 (2005), pages 807–815.

[7] K. Mayer, R. Marklein, K.-J. Langenberg, and T. Kreutter. "Three dimensional imaging system based on Fourier transform synthetic aperture focusing technique". In: *Ultrasonics* 28 (1990). DOI: 10.1016/0041-624X(90)90091-2.

[8] G. A. McMechan. "Migration by extrapolation of time dependent boundary values". In: *Geophys. Prospect* 31 (1983). DOI: 10.1111/j.1365-2478.1983.tb01060.x.

[9] S. Müller, E. Niederleithinger, and T. Bohlen. "Reverse Time Migration: A Seismic Imaging Technique Applied to Synthetic Ultrasonic Data". In: *International Journal of Geophysics* (2012). DOI: 10.1155/2012/128465.

[10] H.-W. Reinhardt. "Echo-Verfahren in der Zerstörungsfreien Zustandsuntersuchung von Betonbauteilen". In: *Beton-Kalender 2007*. Wiley-VCH Verlag GmbH, Apr. 2014, pages 479–595. DOI: 10.1002/9783433600696.ch5.

[11] M. Schickert, M. Krause, and W. Müller. "Ultrasonic Imaging of Concrete Elements Using Reconstruction by Synthetic Aperture Focusing Technique". In: *J. Mater. Civ. Eng.* 15.3 (2003). DOI: 10.1061/(ASCE)0899-1561(2003)15:3(235).

Augmenting Databases-as-a-Service with Policy-Support

Max Plauth, Felix Eberhard, and Andreas Polze

Hasso Plattner Institute, Potsdam, Germany
{firstname.lastname}@hpi.de

Cloud computing offers the potential to store, manage, and process data in highly available, scalable, and elastic environments. Yet, these environments still provide very limited and inflexible means for customers to control their data. For example, customers can neither specify security of intercloud communication bearing the risk of information leakage, nor comply with laws requiring data to be kept in the originating jurisdiction, nor control sharing of data with third parties on a fine-granular basis. This lack of control can hinder cloud adoption for data that falls under regulations. As a part of our efforts in the *Scalable and Secure Infrastructures for Cloud Operations* (SSICLOPS) project, our work during the *Spring 2017* period was focused on implementing a prototypical approach that facilitates policy adherence support in the Hyrise-R in-memory research database.

1 Introduction

As a part of our efforts in the *Scalable and Secure Infrastructures for Cloud Operations* (SSICLOPS) project, we are investigating the technical implications of employing policy language concepts discussed in [1] by example of the use case scenario illustrated in Figure 1. The scenario includes numerous users, where each user requests an instance of the *Hyrise-R* in-memory database in a *Platform as a Service* (PaaS) like manner. However, users impose certain requirements regarding attributes ranging from the coarse-grained properties such as data center location to fine-grained requirements like database configuration parameters. The *policy decision point* (PDP) acts as the main entry point for users requests. While Figure 1 depicts the *PDP* as a centralized component, its actual implementation strategy might vary. Based on the policies specified by a user, the *PDP* routes requests through a series of *policy enforcement points* (PEP) in order to comply with the respective policies. With policy language concepts at hand, users can impose requirements on service providers by annotating their requests accordingly. On the coarse level, requirements such as geolocation or *Quality-of-Service* (QoS) might be expressed, whereas the fine-grained level can be used to specify application specific demands like availability properties or user rights.

The scenario demonstrates that multiple components have to cooperate in order to consider policy requirements on all levels. Over the past FutureSOC periods, we focused on general design aspects of policy-aware cloud stacks and evaluated implementation strategies for facilitating policy support at the infrastructure layer in OpenStack using minimally invasive changes [5]. In this period, we focused on aug-

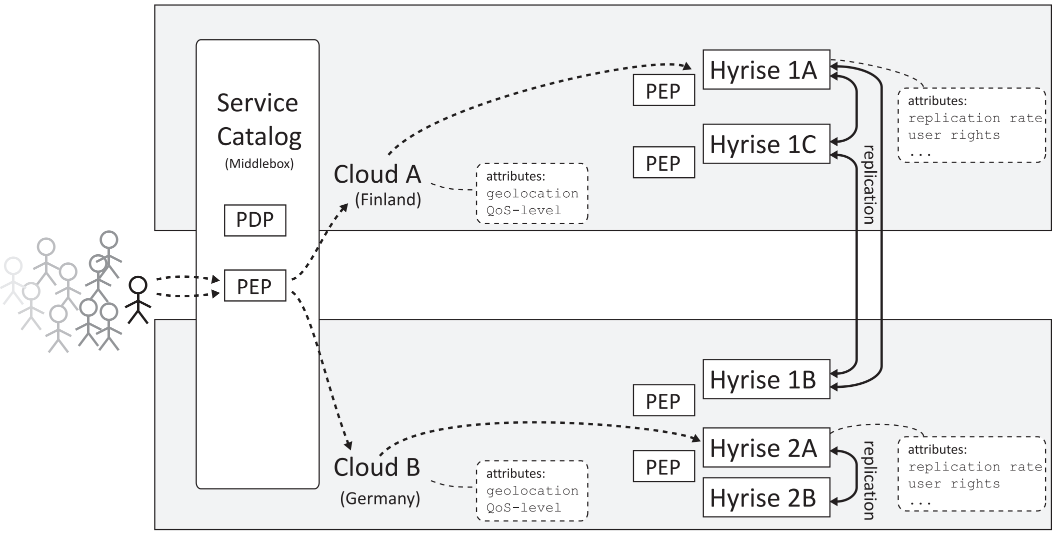


Figure 1: Use case scenario: Users request instances of the *Hyrise-R* in-memory database and annotate their requests with certain policy demands. The *policy decision point* (PDP) acts as the initial entry point and routes requests through a series of *policy enforcement points* (PEP) to process the requests accordingly.

menting the application layer with policy support, where the Hyrise-R in-memory research database has been used as an exemplary application.

2 Motivation: Databases as a Service

Database administration is known to be a demanding task, since expert knowledge is required to properly set up and tune databases to provide good performance. Hence, outsourcing the operation of databases to corresponding PaaS offerings is becoming increasingly popular. Especially for PaaS-based database offerings, strict policy adherence is vital, as databases often hold crucial business assets. To not impede the substantial performance gains of *In-Memory Databases* (IMDB), it is necessary that policy adherence mechanisms do not tax the overall performance of PaaS-based *IMDB* offerings.

3 Approach

To study and enable efficient realization of policy support in *IMDBs*, we augmented the Hyrise [2] open source in-memory research database with policy adherence mechanisms based on CPPL [3]. Hyrise uses replication mechanisms to support cloud-based scale-out deployment [6], elasticity [4], as well as high availability features. Currently, Hyrise-R implements full replication, where the entire data is stored at every node in the database cluster. Using this approach, every node can process

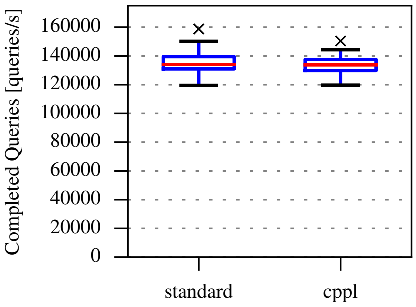


Figure 2: Performance of policy support in Hyrise-R: For high-throughput, transactional workloads, the policy evaluation incurs no notable overhead.

every query. However, expressive policy adherence mechanisms enable more fine-grained replication and data distribution mechanisms in cloud scenarios compared to maintaining full copies. For example, some data may be restricted to be stored in specific geographic locations. Other data may demand a minimum replication rate for high availability. In these cases, partial replication enables increased flexibility and improves resource utilization, as every database node only has to maintain a subset of the entire database. Using partial replication, a policy annotation is crucial, if the user wants to specify what data fragments are allowed to be stored on which nodes.

As an initial prototype on the way towards achieving partial replication, we implemented policy adherence support for the properties location and replication rate based on the CPPL policy language. First, we adapted the query dispatcher to only forward queries to replica nodes located in a permissible location, storing all queried data fragments. Second, integrating policy support also affected the synchronization mechanism of replicas, i.e., the transmission of data manipulation messages only to permissible nodes storing related data fragments.

4 Performance evaluation

Performance measurements were conducted by simulating a high-throughput transactional workload. No-op queries were used instead of actual operations to exclude confounding effects of actual operations as potential decelerating components that do not contribute to the integration of CPPL. To retrieve a sufficiently meaningful dataset, we performed 30 repeated measurements of both the standard and the CPPL-enabled implementations of the Hyrise-R dispatcher, which is backed by two Hyrise replica instances. The results depicted in Figure 2 indicate negligible overhead for policy evaluation. Thus, based on the compact representation and the efficient matching of policies in CPPL, cloud-optimized scale-out deployments of in-memory databases can offer policy support without notable performance degradation.

5 Outlook

Our goal for the upcoming *Fall 2017* period of the Future SOC Lab is to evaluate secure enclave mechanisms in a federated OpenStack testbed. In addition to evaluating general characteristics of trusted computing approaches, we are also planning to implement a prototypical key-value store for handling sensitive data in secure enclaves.

References

[1] F. Eberhardt, J. Hiller, S. Klauck, M. Plauth, A. Polze, and K. Wehrle. *D2.2: Design of Inter-Cloud Security Policies, Architecture, and Annotations for Data Storage*. Technical report. Jan. 2016.

[2] M. Grund, J. Krüger, H. Plattner, A. Zeier, P. Cudré-Mauroux, and S. Madden. "HYRISE - A Main Memory Hybrid Storage Engine". In: *PVLDB* 4.2 (2010), pages 105–116.

[3] M. Henze, J. Hiller, S. Schmerling, J. H. Ziegeldorf, and K. Wehrle. "CPPL: Compact Privacy Policy Language". In: *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society*. WPES '16. Vienna, Austria: ACM, 2016, pages 99–110. ISBN: 978-1-4503-4569-9. DOI: 10.1145/2994620.2994627.

[4] S. Klauck. "Scalability, Availability, and Elasticity through Database Replication in Hyrise-R". In: *Proceedings of the Fourth HPI Cloud Symposium "Operating the Cloud"*. 2016.

[5] M. Plauth, M. Bastian, and A. Polze. "Facilitating Policy Adherence in Federated OpenStack Clouds with Minimally Invasive Changes". In: *Proceedings of the Fifth HPI Cloud Symposium "Operating the Cloud"*. 2017.

[6] D. Schwalb, J. Kossmann, M. Faust, S. Klauck, M. Uflacker, and H. Plattner. "Hyrise-R: Scale-out and Hot-Standby through Lazy Master Replication for Enterprise Applications". In: *Proceedings of the 3rd VLDB Workshop on In-Memory Data Mangement and Analytics*. ACM, 2015, 7:1–7:7. ISBN: 978-1-4503-3713-7. DOI: 10.1145/2803140.2803147.

The relation between the infrastructure-stressing quality and client's geodemographic segment, report for the work carried on the HPI premises

Julia Sidorova[1], Sasidhar Podapati[1], Lars Lundberg[1], and Lars Sköld[2]

[1] Blekinge Institute of Technology
lars.lundberg@bth.se
[2] Telenor Swerige AB
Lars.Skold@telenor.se

The population in Sweden is growing rapidly due to immigration. In this light, the issue of infrastructure upgrades to provide telecommunication services is of importance. New antennas can be installed at hot spots of user demand, which will require an investment, and/or the clientele expansion can be carried out in a planned manner to promote the exploitation of the infrastructure in the less loaded geographical zones. In this paper, we explore the second alternative. Informally speaking, the term Infrastructure-Stressing describes a user who stays in the zones of high demand, which are prone to produce service failures, if further loaded. We have not been able to generalize a decision boundary between the IS and non-IS customers in the feature space defined by the features available at the moment when a new client applies to become a customer, but instead we have studied the Infrastructure-Stressing population in the light of their correlation with geodemographic segments.

1 Introduction

In the era of big data a mapping is desired from multitudes of numeric data to a useful summary and insights expressed in a natural language yet with a mathematical precision [11]. Fuzzy logic bridges from mathematics to the way humans reason and the way the human world operates. Clearly, the "class of all real numbers which are much greater than 1," or "the class of beautiful women," or "the class of tall men," do not constitute classes or sets in the usual mathematical sense of these terms. Yet, "the fact remains that such imprecisely defined notions play an important role in human thinking, particularly in the domains of decision-making, abstraction and communication of information" [13]. Few works exist in business intelligence that use fuzzy logic due to certain inherent difficulties of creating such applications, and yet; despite them, such applications are possible and very useful, e.g. the reader can be referred to a review [3]. Our idea is neither of the two, and it aims to implement the above mentioned insight by Zadeh about completing a useful summary from multitudes of data. Fuzzy logic enables us to formulate a natural language interface between big data, numeric analytics, and a manager, hiding the compexity of data and methods and providing him/her with a comprehensible summary. We summarize data using linguistic hedges (very, rather, highly) and formulate queries such as *"Tell me which neighbourhoods are safe to target, if I want more clients but my infrastructure is highly loaded"*. *"Tell me, whether the infrastructure is rather loaded or highly loaded in the region."* Our specific application is targeting different user segments to fill in the spare capacity of the network in a network-friendly manner. In [7], the notion of *Infrastructure-Stressing* (IS) Client was proposed together with the method to reveal such clients from the customer base. Informally, IS clients use the infrastructure in a stressing manner, such as always staying in the zones of high demand, where the antennas are prone to service failures, if further loaded. Being IS is not only a function of the user's qualities, but also of the infrastructure, and of the relative mobility of the rest of the population.

In the previous HPI proposal we formulated an idea to predict the IS client from the available personal features unrelated to mobility. The results have not been satisfactory (60 % on a two class problem). Instead we have studied a relation between the IS property and geodemographic segments. A more complete account of this work can be found in [4].

For marketing campaigns geodemographic segmenations (like ACORN or MOSAIC) are used, since it is known how the segments can be targeted to achieve the desired goal, as for example, the promotion of a new mobile service in certain neigbourhoods. The client's home address determines the geodemographic category. People of similar social status and lifestyle tend to live close [1, 6]. Geodemographic segmentation provides a *collective view point*, where the client is seen as a representative of the population who live nearby. However, in recent research, it has been shown that the problem of resource allocation in the zones with nearly overloaded and underloaded antennas is better handled relying on *individual modelling* based on the client's historical trajectories [5]. The authors completed a user segmentation based on clustering of user trajectories and it was demonstrated that network planning is more effective, if trajectory-based segments are used instead of geo-demographic segments. Our aim is to explore the ways to connect the individual trajectory-based view on IS customers and the geodemographic view in order to devise analytics capable to complete the efficient analysis based on the individual view point and yet be useful in marketing campaigns in which geodemographic groups are targeted. As a practical conclusion, we have compiled a ranked list of the segments according to their propensity to contain IS clients and crafted two queries:

1. Which segments contain a low or moderate number of IS clients? (target them, while the infrastructure is still rather underloaded)

2. Which segment is highly devoid of IS clients? (target them, when the customer base becomes mature and the infrastructure becomes increasingly loaded).

The simulation of the resulting fuzzy recommendations guarantees the absence of false negatives, such as, concluding that certain segments are safe to hire from, but in fact that would lead to a service failure at some place in the network.

The rest of the paper is organised as follows. Section 2 describes the data set. In Section 3 the proposed methodology is explained. In Section 4, the experiments are reported, and finally the conclusions are drawn and discussion is held in Section 5.

2 Data

The study has been conducted on anonymized geospatial and geodemographic data provided by a Scandinavian telecommunication operator. The data consist of CDRs (Call Detail Records) containing historical location data and calls made during one week in a mid-size region in Sweden with more than one thousand radio cells. Several cells can be located on the same antenna. The cell density varies in different areas and is higher in city centers, compared to rural areas. The locations of 27010 clients are registered together with which cell serves the client. The location is registered every five minutes. During the periods when the client does not generate any traffic, she does not make any impact on the infrastructure and such periods of inactivity are not included in the resource allocation analysis. Every client in the database is labeled with her MOSAIC segment. The fields of the database used in this study are:

- the cells IDs with the information about which users it served at different time points,
- the location coordinates of the cells,
- the time stamps of every event (5 minute resolution),
- the MOSAIC geodemographic segment for each client, and
- the Telenor geodemographic segment for each client.

There are 14 MOSAIC segments present in the database. The six in-house Telenor segments were developed by Telenor in collaboration with InsightOne, and, to our best knowledge, though not conceptually different from MOSAIC, they are especially crafted for telecommunication businesses.

3 A Link between IS and Geodemographic Segments

3.1 Notation and Definitions of Fuzzy Logic

Definition (in the style of [13])**.** A fuzzy set A in X is characterized by a membership function $f_A(x)$, which associates with each point in X a real number in the interval $[]0,1]$, with the value of $f_A(x)$ at x representing the "grade of membership" of x in A. For the opposite quality: $f_{notA}(x) = 1 - f_A(x)$.

Fuzzy membership scores reflect the varying degree to which different cases belong to a set. Under the six value fuzzy set, there are six degrees of membership 1:

fully in, $[0.9-1)$: mostly but not fully in, $[0.6-0.9)$: more or less in, $[0.4-0.6)$: more or less out, $[0.1-0.4)$: mostly but not fully out, $[0-0.1)$: fully out. For a comprehensive guide of good practices in fuzzy logic analysis in social sciences the reader is referred to, for example, [8].

3.2 Linguistic Hedges:

- *Rather* will be added to a quality A, if the square root of its membership function $f_A(x)^{1/2}$ is close to 1.
- *Very* will be added to a quality A, if the square of its membership function $f_A(x)^2$ is close to 1.
- *Extremely* will be added to a quality A, if $f_A(x)^3$ is close to 1.

The principles for calculating the values of hedged membership functions, for example $f_{veryA}(x) = f_A(x)^2$, are described in [4]. Then, given the new membership function, the same principle applies: the closer to 1, the higher is the degree of membership.

3.3 Query Formulation

To keep the formulations and questions naturally sounding, the word infrastructure-friendly (IF) is used. The quality IF is defined as the opposite to IS: $f_{IF}(segment_i) = 1 - f_{IS}(segment_i)$, for some segment *i*. As mentioned above, within the same geodemographic segment, the clients differ with respect to the degree of being IS. When the infrastructure is not overloaded, that is, the recent historical load is still significantly smaller than the capacity, then virtually any client is welcome. As the infrastructure becomes more loaded, the operator wants to be more discriminative. We define being "loaded" for an antenna as a fuzzy variable:

$$f_{loaded}(\text{antenna j}) = max_{\text{all t}}\{load(j,t) \times \text{capacity}(\text{antenna j})^{-1}\}.$$

This quality is measured in man units. Being loaded for infrastructure is defined as:

$$f_{loaded}(\text{infrastructure}) = max_{\text{all antennas j}}\{f_{loaded}(\text{antenna j})\}.$$

Since being loaded is a dangerous quality, we set the strength of the system to be equal to the strength of its weakest component, and for this reason the equation above we use the max operator.

3.4 Queries:

1. Which segments to target, provided that *rather* IF users are acceptable clientele?
2. Which segments to target, provided that only *very* IF are wanted?

Depending on the load, there are different rankings of segments. If initially some segments were in the same tier, for example, "very IF segments", some of them fall out of the tier, as the hedge operator is applied and the value of the membership function is squared (for "extremely IF"). The context, when to apply Query 1 or 2, becomes clarified via calculating f_{loaded} (infrastructure) and checking the applicability of different hedges. The method to obtain fuzzy heuristics is summarized to the sequence of the following steps.

Step 1: The IS clients in the customer base are revealed with the method [7] (the algorithm is reproduced as function *reveal_IS* clients in Algorithm 1), and each client is labeled with the IS/notIS descriptor.

Step 2: The propensity of a segment to contain IS clients is defined as the frequency of IS clients among its members and it is calculated from the data:

$$f_{IS}(segment_i) = frequency_{IS}(segment_i).$$

For linguistic convenience the term Infrastructure- Friendly (IF) is introduced and is set to be opposite to IS:

$$f_{IF}(segment_i) = 1 - f_{IS}(segment_i).$$

Step 3: The ranking of segments is carried out with respect to their IF quality and the hedged values of the membership function are calculated: for all segments i, $f_{ratherIF}(segment_i), f_{veryIF}(segment_i)$, and $f_{extremelyIF}(segment_i)$. Given a hedge, which also codes the severity of the context, the segments fall into the different tiers (corresponding to one of the six fuzzy values): "fully in", "mostly but not fully in", "more or less in", and so on.

Step 4: Locally for the region under analysis, the infrastructure is assessed as *loaded*, *very loaded*, or *extremely loaded*, and thus the severity of the context is assessed. A ranking from Step 3 corresponding to a particular hedge is selected (as a leap of faith further verified in the next section).

The above is depicted as a flow chart in Figure 1.

3.5 Query Simulation

In the above, when deciding which context should be applied, we relied on an intuitive rule: If the infrastructure is $< hedge >$ loaded, then $< hedge >$ IS segments are suitable to hire clients from. For example, in the case of a *rather* loaded infrastructure, *rather* IS segments are suitable targets. Given the expected success of the campaign, e.g. the campaign can attract 300 new clients or 1500 clients, it is possible to simulate the impact of the expected result on the infrastructure. A warning is thrown, if some antenna is overloaded, i.e. when the expected footprint by the segment violates a restriction for some segment i, at some antenna j, some time moment t:

$$\alpha S_{i,j,t} \leq C_j,$$

where α is a scaling coefficient:

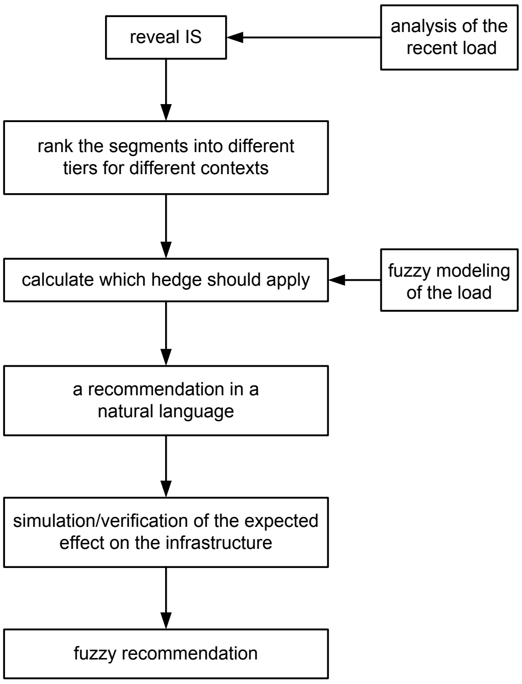

reveal IS
analysis of the recent load
rank the segments into different tiers for different contexts
calculate which hedge should apply
fuzzy modeling of the load
a recommendation in a natural language
simulation/verification of the expected effect on the infrastructure
fuzzy recommendation

Figure 1: The flow chart for the calculation of fuzzy recommendation for a marketing campaign.

$$\alpha = \text{expected number of new clients} \times (\text{current number of clients})^{-1}.$$

This is a justifiable approximation, since there is a high predictability in user trajectories within different segments, e.g. [2, 12].

4 Experiment

For more clarifications about the experiment, the reader is referred to [4]. Due to page limitation we can only sketch the experimental work.

1. **Reveal the IS clients.** Applying the algorithm to reveal IS clients [9], we have added a field to the data set with the label IS or not IS as a descriptor for each client.
2. **Calculate degree of infrastructure- friendliness for each segment.** In the whole customer base, 7 % of subscribers were revealed to be IS [7]. We have obtained the distribution of the IS clients within the MOSAIC and Telenor segments and depicted them in Figure 2 and 3, respectively.

II. Reasoning behind the queries. Each of the 14 MOSAIC classes qualifies as *rather IF*, which are those with $f_{IF}(i)^{1/2} > 0.9$. Once the customer base becomes

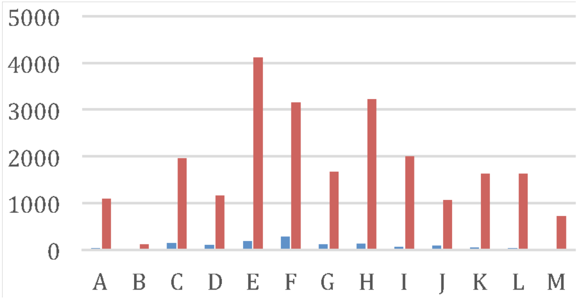


Figure 2: The number of IS clients in different MOSAIC categories

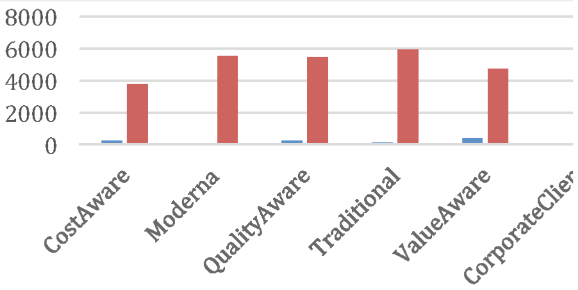


Figure 3: The number of IS clients in different Telenor segments

larger and the spare capacity diminishes, only *very IF* will be wanted, which are those with $f_{IF}(i)^2 > 0.9$. Out of those, only 9 segments qualify as *very* IF and five segments qualify as *extremely* IF $f_{IF}(i)^3 > 0.9$. The customer population was subjected to the same analysis with respect to Telenor segmentation. Each of the six Telenor segments is rather friendly, and there are four and three very and extremely IF segments, respectively.

5 Future Directions and Conclusions

An implicit assumption in our previous research [4, 6, 9, 10] is that we left the load and the tariffs paid by the customers as open variables. Now we will integrate this information and extend the model which is currently based on user trajectories only. Telenor will provide us with a new database with the information on the actual data consumption, we will modify our model and the corresponding scripts and execute them on the HPI machines.

Apart from the experimental research, this term we plan to both continue working on a deeper theoretical model and do outreach. There is a societal aspect to our work. The population of Sweden is growing, the country is running a risk to become short of police, hospitals, and so on. We will work on the question from the societal view point: how to allocate people and what it means for businesses.

When it comes to designing strategies of accomodating many more clients, being IS-prone for a segment is an important quality. We have not been able to generalize a decision boundary between the IS and non-IS customers in the feature space representing their purchasing history and other features available at the moment when a new client applies to become a customer. We have studied the correlation between IS users and the geo-demographic segments, motivated by the fact that we can target the geodemographic segments (MOSAIC and Telenor) in marketing campaigns. For different contexts, we have completed candidate rankings of geo-demographic segments, and, given the absence of other preferences, the top-tier segments are preferable. Which ranking out of several candidate ones is taken depends on the hedge calculated for the intensiveness of infrastructure exploitation. The simulation of the expected effect guarantees no false negatives, such as saying that certain segments are safe to hire from, but in fact that would lead to a service failure at some place and time in the network.

References

[1] J. Debenham, G. Clarke, and J. Stillwell. “Extending geodemographic classification: a new regional prototype”. In: *Environment and Planning A* 35.6 (2003), pages 1025–1050. DOI: 10.1068/a35178.

[2] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson. “Approaching the limit of predictability in human mobility”. In: *Scientific reports* 3 (2013).

[3] A. Meyer and H. J. Zimmermann. “Applications of fuzzy technology in business intelligence”. In: *International Journal of Computers Communications & Control* 6.3 (2011), pages 428–441. DOI: 10.15837/ijccc.2011.3.2128.

[4] S. Podapati, L. Lundberg, L. Skold, O. Rosander, and J. Sidorova. “Fuzzy Recommendations in Marketing Campaigns”. In: *IISA 2017 The 8th IEEEh International Conference on Information, Intelligence, Systems and Applications. 28-30 August, Larnaca, Cyprus*. 2017. DOI: 10.1007/978-3-319-67162-8_24.

[5] C. C. Ragin. “Qualitative Comparative Analysis using Fuzzy Sets (fsQCA)”. In: *Configurational Comparative Methods: Qualitative Comparative Analysis (QCA) and Related Techniques*. SAGE Publications, Inc., 2009, pages 87–122. DOI: 10.4135/9781452226569.n5.

[6] S. Sagar, L. Skold, L. L., and S. J. “Trajectory Segmentation for a Recommendation Module of a Customer Relationship Management System”. In: *The 2017 Int. Symposium on Advances in Smart Big Data Processing*. 2017. DOI: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.177. Forthcoming.

[7] J. Sidorova, L. Skold, and L. L. *HPI Future SOC Lab: Proceedings 2016*. Edited by C. Meinel, A. Polze, G. Oswald, R. Strotmann, U. Seibold, and B. Schulzki. Chapter Revealing Infrastructure-Stressing Clients in the Customer Base of a Scandinavian Operator using Telenor Mobility Data and HPI Future SoC Lab Hardware Resources.

[8] J. Sidorova, L. Skold, O. Rosander, and L. L. “Discovering insights in telecommunication business from an interplay of geospatial and geo-demographic factors”. In: *The 1st International Workshop on Data Science: Methodologies and Use-Cases (DaS 2017) at 21st European Conference on Advances in Databases and Information Systems (ADBIS 2017), 28-30 August, Larnaca, Cyprus*. LNCS. In press.

[9] J. Sidorova, O. Rosander, L. Skold, H. Grahn, and L. Lundberg. “Finding a Healthy Equilibrium of Geo-demographic Segments for a Telecom Business: Who Are Malicious Hot-Spotters?” In: *Machine Learning Paradigms*. Springer International Publishing, July 2018, pages 187–196. DOI: 10.1007/978-3-319-94030-4_8.

[10] J. Sidorova, L. Skold, O. Rosander, and L. Lundberg. “Recommendations for Marketing Campaigns in Telecommunication Business based on the footprint analysis”. In: *The 1st International Workshop on Data Science: Methodologies and Use-Cases (DaS 2017) at 21st European Conference on Advances in Databases and Information Systems (ADBIS 2017), 28-30 August, Larnaca, Cyprus*. LNCS.

[11] L. Zadeh. “Fuzzy Logic and Beyond - A New Look”. In: *Fuzzy Sets and their applications to cognitive and decision processes: Proceedings of the US-Japan seminar on fuzzy sets and their applications, held at university of California, Berkeley, California, July 1-4, Academic Press*. Edited by L. Zadeh, F. King-Sun, and T. Konichi. 2014.

[12] L. A. Zadeh. *A fuzzy-set-theoretic interpretation of linguistic hedges*. 1972.

[13] L. A. Zadeh. “Fuzzy sets”. In: *Information and control* 8.3 (1965), pages 338–353. DOI: 10.1016/S0019-9958(65)90241-X.

The Structure of Industrial SAT Instances

Tobias Friedrich, Ralf Rothenberger, and Andrew M. Sutton

Hasso Plattner Institute, Potsdam, Germany
{tobias.friedrich,andrew.sutton}@hpi.de

A vast number of combinatorial problems can be encoded as propositional formula over a set of Boolean variables. A solution to the problem reduces to finding an assignment to these variables that satisfies the formula. Despite negative worst-case complexity results, many large industrial SAT instances can be solved efficiently by modern solvers. The goal of this ongoing project is to study the structure and hardness of different instances of propositional satisfiability. Our aim is to determine what properties are essential for efficient SAT solving.

1 Introduction

Propositional satisfiability (SAT) is one of the most fundamental problems in computer science. Many practical questions from different domains can be encoded as propositional formula and solved by determining the satisfiability of the resulting formula. A propositional formula is constructed from a set V of n Boolean variables by forming a conjunction

$$F = C_1 \wedge C_2 \wedge \cdots \wedge C_m$$

of m disjunctive clauses where

$$C_i = (\ell_1 \vee \ell_2 \vee \cdots \vee \ell_{k_i}).$$

where $\ell_j \in \{v, \neg v\}$ for some $v \in V$. Here $\neg v$ denotes the logical negation of v. The goal of the decision problem is to decide if there is an assignment to all variables of V so that F evaluates to true. SAT is a central problem in theoretical computer science, but it is also an important practical problem since many difficult combinatorial problems reduce to it.

SAT instances and distributions

A distribution of SAT instances is typically parameterized by n and m and is described by a categorical distribution over all formulas over n variables and m clauses. The most heavily studied distribution of SAT instances is the *uniform* distribution. The uniform distribution is the distribution $U_{n,m}$ of all well-formed CNF formulas on n variables and m clauses where each formula has the same probability of being selected.

Most theoretical work on SAT instances has focused almost exclusively on this uniform distribution $U_{n,m}$. Uniform random formulas are easy to construct, and have shown to be accessible to probabilistic analysis due to their statistical uniformity. Indeed, a long line of successful research has relied on the uniform distribution, and from it, several sophisticated rigorous and non-rigorous techniques have developed for analyzing random structures in general.

Nevertheless, a focus on uniform random instances comes with a risk of driving SAT research in the wrong direction [9] because such instances do not possess the same structural properties as ones encountered in practice. It is well-known that solvers that have been tuned to perform well on one class of instances do not necessarily perform well on another [3], and studying the algorithmics of solvers on uniform random formulas can lead research astray.

The empirical SAT community has expanded their view to study *industrial* instances. Industrial instances arise from problems in practice, such as hardware and software verification, automated planning and scheduling, and circuit design. Empirically, industrial instances appear to have strongly different properties than formulas generated uniformly at random, and as might be expected, SAT solvers behave very differently when applied to them [6, 10].

Furthermore, a number of *non-uniform* random distributions have been recently proposed. These models include regular random [4], geometric [5] and scale-free [1, 2]. The scale-free model is especially promising because the *degree distribution* (distribution of variable occurrence) of instances follows a power-law and this phenomenon has been observed on real-world industrial instances.

Project aim. This project is an ongoing effort to study the influence of formula structure on hardness, specifically comparing traditional SAT solvers. We aim to utilize the parallel computing power of the 1000 node cluster of the Future SOC Lab to sample random distributions of formulas by generating a massive set of large random formulas and run a SAT solver on them to check their satisfiability and hardness.

2 Hardness of scale-free formulas

In previous terms of the Future SOC Lab, we studied *scale-free* instance distributions. In particular, a scale-free formula F on n variables and m clauses of length k is constructed as follows. We define a set of n weights

$$w_i := \left(\frac{n}{i}\right)^{\frac{1}{\beta-1}}, \quad \text{for each } i \in [n],$$

where $\beta > 0$ is a parameter called the *power-law exponent*. Let $V = \{v_1, v_2, \dots, v_n\}$ denote the set of variables. Rather than picking each variable uniformly at random to construct a clause, we select variable v_i with probability

$$p_i = \frac{w_i}{\sum_{j=1}^{n} w_j}.$$

Each of the m clauses is then sampled independently at random using $\{p_i \colon i \in [n]\}$ to sample the variables as follows:

1. Select k variables independently at random from the distribution $\{p_i : i \in [n]\}$. Repeat until no variables coincide.
2. Negate each of the k variables independently at random with probability $1/2$.

Thus each such formula is generated at random, but the resulting degree distribution follows a power-law with exponent β.

The *constraint density* of a formula is the number of clauses divided by the number of variables. We are interested in locating the *phase transition* of the scale-free model. Specifically, we want to know for each value of power-law exponent β, what is the constraint density where the probability that a formula is satisfiable reaches $1/2$. We conjecture this corresponds to formulas that are difficult to solve by SAT solvers.

We employed the Future SOC lab cluster for sampling formulas from scale free distributions in parallel, and measuring the running time of SAT solvers to determine where the hardest formulas are with respect to constraint density and power-law exponent. We generated a number of scale-free formulas and checked them using SAT solvers. We used GNU Parallel [11] to distribute a large number of jobs over the cluster. Each job was responsible for generating a set of random scale-free formulas. Each formula was then solved by MapleCOMSPS, a state-of-the-art SAT solver. For each power-law exponent, and each constraint density value, we measured the satisfiability of the formula, along with the amount of time needed by the solver. This gives us a preliminary picture of the difficulty of formulas with respect to both constraint density and power-law exponent. For the remainder of the term, we will focus on these results, generating them for larger formulas and determining empirically the exact threshold as a function of power-law exponent and constraint density.

We plot the results from our FSOC experiments. The proportion of satisfiable formulas as a function of constraint density for a representative set of formulas with $n = 700$ and various power-law exponent values is shown in Figure 1. Note that as β increases, the transition from soluble to insoluble phase shifts to the right (higher constraint densities). Increasing β values further, it seems evident that the empirical threshold is approaching the supposed critical density of the uniform random 3-SAT model, i.e., $r_3 \approx 4.26$.

In Figure 2 we show the estimate of the threshold as a function of β for $n = 700$. The broken line illustrates the actual proportion of satisfiable instances at the proposed threshold. Figure 3 illustrates the effect of n on the threshold at different values for the power-law exponent. As expected, the threshold appears to converge to a constant for fixed β and $n \to \infty$. Moreover, the empirical values seem to be already close to the limit for $n = 1000$. Thus, our experiments can serve as a sanity check for future theoretical work on the threshold. Indeed, the experiments conducted on the Future SOC lab cluster have already been useful in aiding the discovery of new theoretical properties about the threshold on the scale-free model [7, 8].

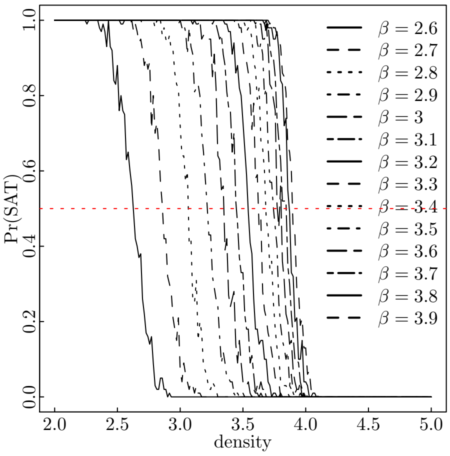


Figure 1: Proportion of satisfiable formulas ($n = 700$) as a function of constraint density m/n for various power-law exponents β

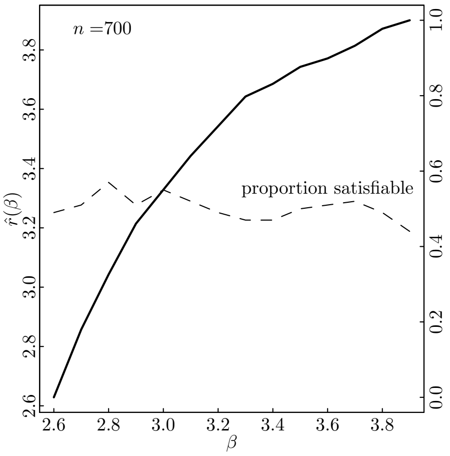


Figure 2: Empirical estimate for critical value $\hat{r}(\beta)$ as a function of power-law exponent β on formulas with $n = 700$ variables. Dashed line denotes the actual proportion of soluble formulas at the proposed threshold.

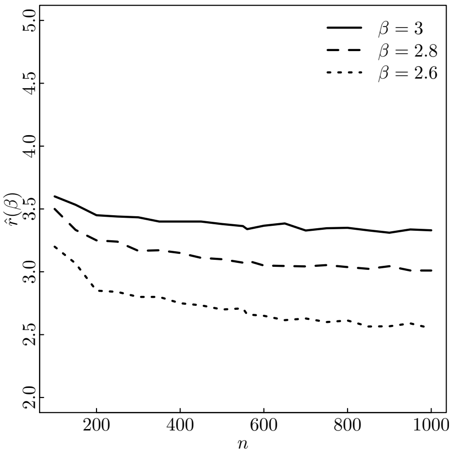


Figure 3: Empirical estimate for critical value $\hat{r}(\beta)$ as a function of problem size n on formulas for various power-law exponent β values

References

[1] C. Ansótegui, M. L. Bonet, and J. Levy. "Towards Industrial-Like Random SAT Instances". In: *21st IJCAI*. 2009, pages 387–392.

[2] C. Ansótegui, M. L. Bonet, and J. Levy. "On the Structure of Industrial SAT Instances". In: *15th CP*. 2009, pages 127–141. DOI: 10.1007/978-3-642-04244-7_13.

[3] M. Birattari. *Tuning Metaheuristics: A Machine Learning Perspective*. Berlin Heidelberg: Springer, 2009. DOI: 10.1007/978-3-642-00483-4.

[4] Y. Boufkhad, O. Dubois, Y. Interian, and B. Selman. "Regular Random *k*-SAT: Properties of Balanced Formulas". In: *9th SAT*. 2006, pages 181–200. DOI: 10.1007/978-1-4020-5571-3_9.

[5] M. Bradonjic and W. Perkins. "On Sharp Thresholds in Random Geometric Graphs". In: *18th Intl. Workshop on Randomization and Computation* (*RANDOM*). 2014, pages 500–514.

[6] J. M. Crawford and A. B. Baker. "Experimental Results on the Application of Satisfiability Algorithms to Scheduling Problems". In: *12th AAAI*. 1994, pages 1092–1097.

[7] T. Friedrich, A. Krohmer, R. Rothenberger, T. Sauerwald, and A. M. Sutton. "Bounds on the Satisfiability Threshold for Power Law Distributed Random SAT". In: *European Symposium on Algorithms* (*ESA*). arXiv:1706.08431. 2017.

[8] T. Friedrich, A. Krohmer, R. Rothenberger, and A. M. Sutton. "Phase Transitions for Scale-Free SAT Formulas". In: *Proceedings of 21st AAAI*. 2017, pages 3893–3899.

[9] H. Kautz and B. Selman. "The state of SAT". In: *Disc. Appl. Math.* 155.12 (2007), pages 1514–1524. DOI: 10.1016/j.dam.2006.10.004.

[10] K. Konolige. “Easy to be hard: Difficult problems for greedy algorithms”. In: *4th KR*. 1994, pages 374–378.

[11] O. Tange. “GNU Parallel - The Command-Line Power Tool”. In: *;login: The USENIX Magazine* 36.1 (Feb. 2011), pages 42–47.

PRIDE-2: Improving Personal Risk Detection Based on One-Class Classifiers and the Use of Wearable Sensors

Jorge Rodríguez[1], Miguel Angel Medina-Pérez[1], Luis A. Trejo[1],
Ari Y. Barrera-Animas[1], Raúl Monroy[1], Armando López-Cuevas[1], and
José Ramírez-Márquez[2]

[1] Tecnologico de Monterrey, México
{jorger,migue,ltrejo,A01373306,raulm,acuevas}@itesm.mx
[2] Stevens Institute of Technology, Hoboken, NJ, USA
jmarquez@stevens.edu

In our previous project [1], we defined personal risk detection as the timely identification of a situation when someone is at imminent peril, such as a health crisis or a car accident. A risk-prone situation should produce sudden and significant deviations in user patterns, and the changes can be captured by a group of sensors, such as an accelerometer, gyroscope, and heart rate monitor, which are normally found in current wearable devices. Previous research findings were published in [1, 8] and presented at HPI Future SOC Lab. The present work rises with the aim of improving our previous results. In order to achieve it, the following two approaches were tested: 1) a visualization method in real-time of PRIDE users leveraged with a one-class classifier called Bagging-TPMiner, 2) the addition of frequency-domain features to the time-domain features embraced in the PRIDE dataset. We reported our visualization model in [5]. Although experiment results reported in this document are encouraging, due to the sheer amount of data, the results presented in this report are partial and considered an ongoing research.

We continued our experiments from last HPI Future SOC Lab period using one-class classifiers over two new sets of features from the publicly available dataset called PRIDE (Personal RIsk DEtection) in order to validate our hypothesis that potential risk situations can be better identified when combining features from both domains: time and frequency. PRIDE is a dataset that contains physiological and behavioral data from 23 users, where each user might reach half a million records.

1 Introduction

The work presented is a natural continuation of previous work performed using a 64-core cluster with 128 GB RAM from HPI Future SOC Lab. Thanks to this support we have been able to report our findings in [1, 8] as mentioned in our technical report for the period that ended on April 2017.

In our previous project, we defined personal risk detection as the timely identification of a situation when someone is at imminent peril, such as a health crisis or a car accident. We worked under the hypothesis that a risk condition produces sudden and significant deviations regarding standard physiological and behavioral user patterns. Monitoring for the occurrence of these changes can be done using a group of sensors, such as the accelerometer, gyroscope, heart rate, etc. Recently we released a dataset, called PRIDE [1] that provides a baseline for the development and the fair comparison of personal risk detection mechanisms.

In this stage of our research, our intention is to improve our previous results. Two approaches were chosen to accomplish our objective. The first one is a novel visualization method, recently published in [5], to track and identify in real-time when a person is in a risk-prone situation. The visualization is leveraged with a traffic light model of one-class classifiers called Bagging-TPMiner, introduced in [7]. Bagging-TPMiner was tested and compared with the classifiers tested in previous work presented at HPI Future SOC Lab Day - Fall 2016. The second approach is the generation of features in the frequency-domain combined with ones in the time-domain, reported in [1]. Our hypothesis here is that an increase in accuracy can be achieved using both types of features.

Regarding the first approach, we propose the development of a Monitor Center (MC) for an individual (patient, citizen, and so on), using data from non-intrusive wearable sensors for each final user. These data is visualized at the Center along with the output of a machine learning algorithm, so a decision maker can react promptly depending on the situation faced by the individual. In the following paragraphs we give more detail about our model, which has been proved to work off-line. Our current proposal has the main goal of implementing the on-line version of our model.

People often face risk-prone situations that range from a mild event to a severe, life-threatening scenario. Risk situations stem from a number of different scenarios: a health condition, a hazard situation due to a natural disaster, a dangerous situation because one is being subject to a crime or physical violence, among others. The lack of a prompt response, calling for assistance, may severely worsen the consequences. We have recently developed a novel visualization method to track and to identify, in real-time, when a person is under a risk-prone situation, such as a health related condition of elderly people. Our visualization model is capable of providing a decision maker a visual description of the physiological behaviour of an individual, or a group thereof; through it, the decision maker may infer whether further assistance is required, if a risky situation is in progress. Our visualization is leveraged with a traffic light model of a one-class classifier. This combination allows us to train the decision maker into visualising correct and potential risky or abnormal behaviour.

Minimising response-time to a risk-prone situation is critical. Depending on the nature and severity of the situation, every minute increases the possibility of a victim suffering severe damage. In fact, the National Service Framework for coronary heart disease stipulates that at least 75 % of category A (emergency) calls should be reached within 8 min. With the growing presence of the Internet of Things (IoT) and the development of wearable devices, the time to assist a person in a potentially damaging situation can be shortened by developing platforms to monitor a subject's behaviour, in real time.

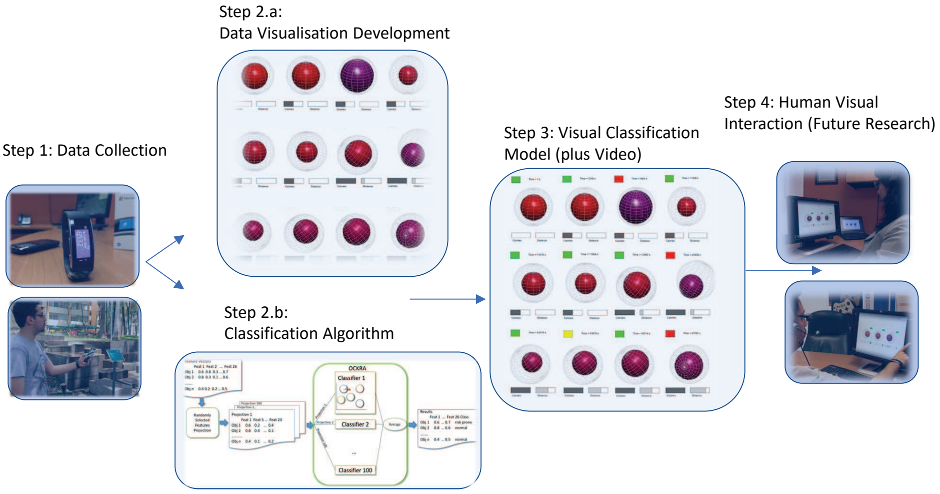


Figure 1: FiToViz model: A visual platform for risk assessment

We have implemented our visualization model into an application, called FiToViz (Fitness to visualization). In FiToViz, our novel visualization model takes as input readings from a set of sensors of a wearable device, and transforms this input into a geometrical, dynamic visual representation. FiToViz provides naturally intuitive and instantaneous feedback about a subject's activity. It may display the visual activity model of one or several users at the same time. Our visualization is leveraged with a traffic light model of a one-class classifier. Given a set of sensor readings, this classifier aims to spot unusual behaviour, reassuring or triggering the decision maker to action. In Figure 1 we provide a general description of the complete process involved in FiToViz. In step 1, measurements from sensors in the user band are collected in order to build a dataset, referred to as PRIDE. Our current visual model works off-line and takes PRIDE as input for step 2, where two processes run simultaneously: During step 2a we developed the visual model from sensor variables, and in step 2b our classification algorithm is trained and tested for every user in the PRIDE dataset. In step 3, our visual model and our classification algorithm are merged, running synchronously and using, for demonstration purposes, a subset of PRIDE from a single user. The following video shows a sample of the outcome: https://youtu.be/fbiHY2B10pM. Step 4, which pictures the work to be developed in this proposal, includes the on-line version of our model, where a decision-maker can assess risk in real time, for a group of users simultaneously. Both the classifier and the visualization model complement each other to help decision making more robust and less error-prone.

Regarding the second approach, we intend to improve our classifiers' performance by using data on the frequency-domain, since our current results are based on time-domain. We first derived the new frequency-domain features to include in our dataset and then followed the next methodology: we reduced dimensionality by applying a correlation analysis and a principal component analysis over time-domain and

Table 1: Description of the Microsoft Band Sensors

Sensor	Description	Frequency
Accelerometer	Provides X, Y, and Z acceleration in g units. 1 g = 9.81 meters per second squared (m/s^2).	8 Hz
Gyroscope	Provides X, Y, and Z angular velocity in degrees per second, (°/s) units.	8 Hz
Distance	Provides the total distance in centimetres, current speed in centimetres per second (cm/s), current pace in milliseconds per meter (ms/m).	1 Hz
Heart Rate	Provides the number of beats per minute, also indicates if the heart rate sensor is fully locked onto the wearer's heart rate	1 Hz
Pedometer	Provides the total number of steps the user has taken.	1 Hz
Skin Temperature	Provides the current skin temperature of the user in degrees Celsius.	33 mHz
UV	Provides the current ultraviolet radiation exposure intensity (None, Low, Medium, High, Very High)	16 mHz
Calories	Provides the total number of calories burned by the user.	1 Hz

frequency-domain features. Then we run our classifiers over the new four subsets of the dataset. In the next sections, we explain in more detail each of the two new approaches and finally we presnet our partial results, which are by far concluding.

2 The PRIDE Dataset

In the current work, we use the personal risk detection (PRIDE) dataset repository, which contains 23 datasets, each one comprised of the records obtained from observing the health measurements of different users, with diverse characteristics regarding gender, age, height, and lifestyle.

PRIDE test subjects are comprised of eight female and 15 male volunteers, aged between 21 and 52 years, with heights between 1.56m and 1.86m, and weights between 42 to 101 kg. The volunteers exercising rates ranged from 0 to 10 hours a week, and the time they spent sitting ranged from 20 to 84 hours a week. The health measurements were done using the sensors on the Microsoft Band v1©, recording the values of the sensors using a mobile application developed using the available SDK, and installed in each user's smartphone. The used sensors from the band and the frequencies of data acquisition for that sensor are described in Table 1.

The data collected from the activities of the user in one week comprises the Normal Conditions Data Set (NCDS), which can be used to construct the normal behavior baseline, which will be used to look for deviations in the behavior, and thus detect risk situations. The same 23 test subjects participated in another data acquisition process to test how the users responded when confronted with an anomalous situation. They needed to perform the following activities: rushing 100 meters as fast as possible,

Table 2: PRIDE feature vector structure (1–18 fields)

Gyroscope Accelerometer						Gyroscope Angular Velocity						Accelerometer					
X axis		Y axis		Z axis		X axis		Y axis		Z axis		X axis		Y axis		Z axis	
$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s	$\bar{x}$	s
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

Table 3: PRIDE feature vector structure (19–26 fields)

Heart Rate	Skin Temperature	Pace	Speed	UV	Δ Pedometer	Δ Distance	Δ Calories
19	20	21	22	23	24	25	26

going up and down the stairs in a multi-floor building as quickly as possible, a two-minute box practice session, falling back and forth, and holding one's breath for as long as possible. Each activity aims to simulate a dangerous or abnormal situation in the real world, e.g., running away from a dangerous situation, evacuating a building during an emergency, defending from an aggressor, swooning, and experiencing breathing problems such as dyspnea. The records of these scenarios comprise the Anomalous Conditions Data Set (ACDS).

Dataset time-domain pre-processing

The PRIDE dataset was preprocessed according to [1], to perform any experiments with one-class classifiers. Tables 2 and 3 show the structure of the 26-dimensional feature vector of each object after the preprocessing phase, where $\bar{\mathbf{x}}$, **s**, and Δ stands for mean, standard deviation, and delta of the value, respectively.

3 Visualization over PRIDE

To support real-time decision making through the use of wearables, we have designed a novel visualization approach proposed in [5]. The visualization consists on mapping the values provided by the sensors of the Microsoft Band v1© into a dynamic image that is both, informative and visually intuitive to the human eye.

The core of the visualization model is formed by two concentric spheres as depicted in Figure 2, where every variable describing the size and position of the spheres is matched to a sensor value. In this sense, the spheres are dynamic in accordance with the variations of the data. In total, the visualization model displays 11 variables. Of these, the inner sphere represents changes of six variables, namely: UV exposure, heart rate, skin temperature, and 3-axis accelerometer. The outer sphere encodes three variables of the 3-axis gyroscope, and two more variables, distance, and calories are encoded as bar graphs.

Regarding the visualization, two videos that show one user activity during a period can be found at the following link: https://youtu.be/07G8HNXvIEc and https:

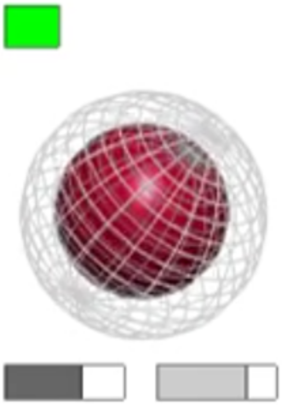


Figure 2: FiToViz model: visualization scheme of concentric spheres encoding sensor data

//youtu.be/m0LZHDTKk5E. Every minute in the video represents one hour of user activity. Extended versions of the video are also available upon request to the authors. Experiments results show that there are several aspects in the visualization that can be naturally assimilated by the decision maker through continuous observation of the visualization; for example, the dynamic changing radius of the inner sphere simulates a heart beating. In Figure 3, four different activities performed by an individual are shown: a) Normal activity; Simulation of a risk situation: b) evacuation alert, c) running away, and d) fighting back. It is important to notice that normal activity differs highly from risk activities.

4 PRIDE in the frequency-domain

The PRIDE dataset was used in its raw form in order to calculate the frequency-domain features and add them to the time-domain features obtained by following the procedure proposed in [1]. The time and frequency-domain features were only calculated for the 3-axis gyroscope and accelerometer sensors. Heart rate, Skin temperature, Delta pedometer, Delta distance, Speed, Pace, Delta calories, and UV features remain as detailed in Table 3. Accordantly, feature vector represents the sensors measurements in an interval of one second.

Frequency-domain features are calculated using the discrete fast Fourier transform [6]. Table 4 shows the frequency and time-domain features calculated for the 3-axis gyroscope accelerometer, gyroscope angular velocity, and accelerometer measurements. Currently, we have 102 time-domain features, 90 frequency-domain features, and eight non-motion features. Thus, a 200-dimensional feature vector represents each one-second observation.

Transforming the PRIDE dataset features to time and frequency-domain is a fist step that will enable us to perform experiments with one-class classifiers. We be-

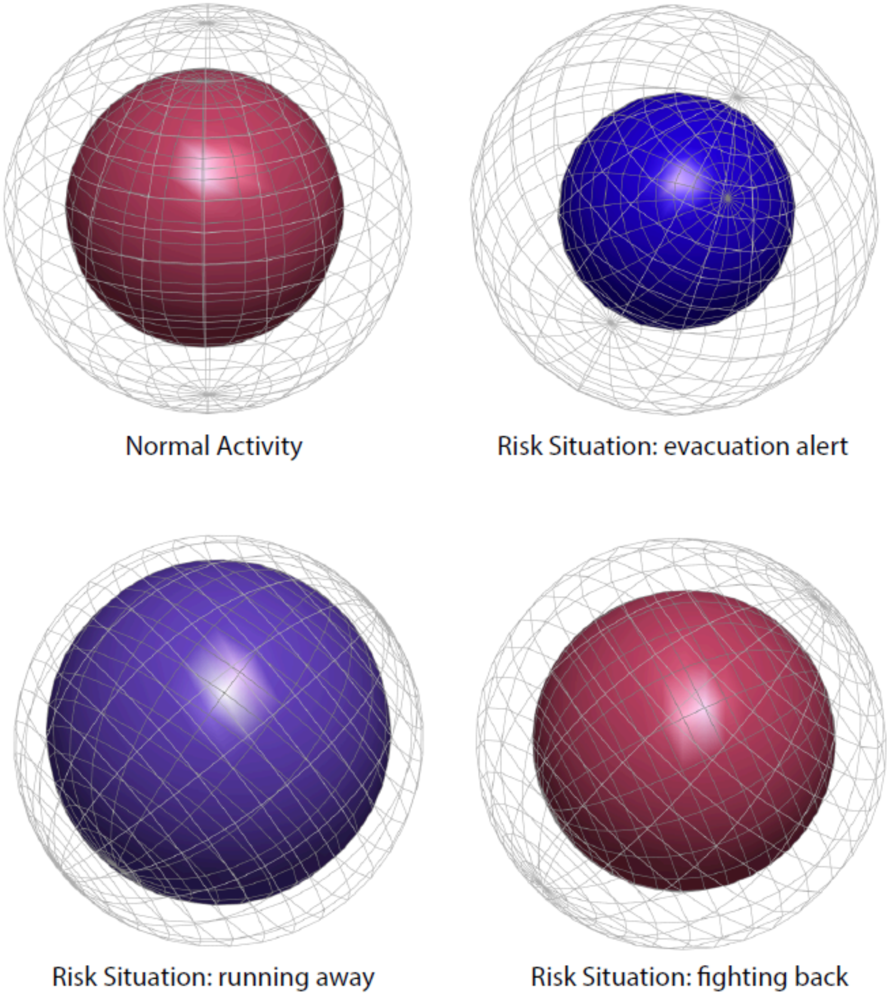

Normal Activity
Risk Situation: evacuation alert
Risk Situation: running away
Risk Situation: fighting back

Figure 3: Four different types of activities visualized: a) Normal activity; Simulation of a risk situation: b) evacuation alert, c) running away, and d) fighting back

Table 4: Features from time and frequency domain of accelerometer and gyroscope sensors

Time-domain	Frequency-domain
Signal magnitude area *	FFT Energy
Root mean square	FFT Mean Energy
Signal Vector Magnitude *	FFT Std Dev Energy
Average Signal Vector Magnitude *	Peak Power
Variance sum *	Peak DFT Bin
Curve length	Peak Magnitude
Average non linear energy	Entropy
Variance	Spectral Entropy
Mean	Peak Frequency
Max	Peak energy
Min	
Standard Deviation	
Median	
Range	

* features calculated with the 3 axes of each sensor

lieve that both types of features will allow us to obtain a better representation of quotidian standard physiological and behavioral user patterns. In consequence, we can increase the accuracy of the classification task of one-class classifiers proposed in [8]. However, we are anticipating an incremental amount of CPU hours due to moving from 26-dimensional feature vectors to 200-dimensional feature vectors, for all 23 users.

Since the number of features increased significantly, we decided to run PCA and a correlation analysis to the new set of features. We present are methodology and results in the next sections.

5 PCA and correlation analysis over PRIDE subset in Time-Domain

We have conducted the principal component analysis (PCA) and correlation matrix method on the PRIDE dataset with the aim to reduce its dimensionality. PCA allows identifying those features that best describe the variability of the data in the dataset. Likewise, the correlation matrix (CM) performs a statistical correlation analysis that are often used to remove redundant (high-correlated) features.

Firstly, we conducted the CM to remove redundant features and then we apply a PCA to remove features that doesn't contribute to the principal components that allows us to retain at least 60 % of data variability.

5.1 Correlation matrix

The execution steps, per user, are:

1. Correlation matrix is computed.
2. Correlation matrix results are plotted and saved in a PDF file.
3. Features with correlation equal or greater than 0.75 are saved to a vector. This vector will include all the highly-correlated features.

Once the correlation matrix was computed for the 23 users, we obtained a frequency table of occurrence of each feature in the 23 previously obtained vectors. From the frequency table, we decided to remove from the dataset the features that were reported by 23 and 22 users as highly-correlated.

5.2 PCA

The execution steps, per user, are:

1. PCA is computed.
2. PCA results are plotted and saved in a PDF file. The file embraces 1) One plot of the percentage of the explained variances across 10 dimensions. 2) One plot per each dimension, with contribution percentage per variable in each dimension. For the TD, 5 dimensions explain (approximately) the 60 % of data variability.
3. PCA results values are saved.

Once PCA was computed for the 23 users, we obtain a frequency table of occurrence of each feature in the corresponding first dimensions of all users. From the frequency table, we decided to remove from the dataset the features that were never used in all first dimensions that retain the 60 % of data variability.

The results in Table 5 show that all users report features 2, 6, and 18 as highly-correlated. Since these features are highly correlated (more than 0.75) with others, they were removed from the PRIDE dataset. Additionally, features 10 and 14 are reported by 22 test-subjects with high correlations with others features. Then, these two features were eliminated due to the number of users that report them. In total, 5 features were removed from the PRIDE TD dataset, reducing its features from 26 to 21. Moreover, the 23 test-subjects report that features 7, 9, 11, 19, 20, 24, and 26 (Average gyro ang vel for axis X, Y, and Z, heart rate, skin temperature, pace, and UV) are below of a correlation value of 0.75.

According with the results in Table 6, 6 features should be removed from the PRIDE TD dataset. However, we make a closer analysis to obtain the frequencies of use of each of these features per dimension across all users. Thus, we can remove features whose contributions to data variability are fewer.

The results in Table 7 show that features 7, 9, 11 and 26 are never used in PCA analysis; that is, these features are not contributing to explain the data variability.

Table 5: Correlation matrix analysis over PRIDE dataset in Time-Domain

Feature number	Frequency reported	Feature name
Feat 1	9	Average Gyro Accel X
Feat 2	**23**	**Std Dev Gyro Accel X**
Feat 3	8	Average Gyro Accel Y
Feat 4	19	Std Dev Gyro Accel Y
Feat 5	15	Average Gyro Accel Z
Feat 6	**23**	**Std Dev Gyro Accel Z**
Feat 7	0	Average Gyro Ang Vel X
Feat 8	2	Std Dev Gyro Ang Vel X
Feat 9	0	Average Gyro Ang Vel Y
Feat 10	**22**	**Std Dev Gyro Ang Vel Y**
Feat 11	0	Average Gyro Ang Vel Z
Feat 12	10	Std Dev Gyro Ang Vel Z
Feat 13	14	Average Accel X
Feat 14	**22**	**Std Dev Accel X**
Feat 15	15	Average Accel Y
Feat 16	21	Std Dev Accel Y
Feat 17	8	Average Accel Z
Feat 18	**23**	**Std Dev Accel Z**
Feat 19	0	Heart Rate
Feat 20	0	Skin Temperature
Feat 21	3	Δ Pedometer
Feat 22	13	Δ Distance
Feat 23	2	Speed
Feat 24	0	Pace
Feat 25	1	Δ Calories
Feat 26	0	Uv

Table 6: Principal component analysis over PRIDE dataset in Time-Domain – part 1

Feature number	Frequency of use in 1-5 dimensions	Feature name
Feat 1	9	Average Gyro Accel X
Feat 2	Removed by CM	Std Dev Gyro Accel X
Feat 3	8	Average Gyro Accel Y
Feat 4	19	Std Dev Gyro Accel Y
Feat 5	15	Average Gyro Accel Z
Feat 6	Removed by CM	Std Dev Gyro Accel Z
Feat 7	**0**	**Average Gyro Ang Vel X**
Feat 8	2	Std Dev Gyro Ang Vel X
Feat 9	**0**	**Average Gyro Ang Vel Y**
Feat 10	Removed by CM	Std Dev Gyro Ang Vel Y
Feat 11	**0**	**Average Gyro Ang Vel Z**
Feat 12	10	Std Dev Gyro Ang Vel Z
Feat 13	14	Average Accel X
Feat 14	Removed by CM	Std Dev Accel X
Feat 15	15	Average Accel Y
Feat 16	21	Std Dev Accel Y
Feat 17	8	Average Accel Z
Feat 18	Removed by CM	Std Dev Accel Z
Feat 19	**0**	**Heart Rate**
Feat 20	**0**	**Skin Temperature**
Feat 21	3	Δ Pedometer
Feat 22	13	Δ Distance
Feat 23	2	Speed
Feat 24	0	Pace
Feat 25	1	Δ Calories
Feat 26	**0**	**Uv**

Table 7: Principal component analysis over PRIDE dataset in Time-Domain – part 2

Feature number	Frequency of use in 1-5 dimensions	Frequency of use across 5 dimensions	Feature name
Feat 1	9	52	Average Gyro Accel X
Feat 2	Removed by CM	Removed by CM	Std Dev Gyro Accel X
Feat 3	8	31	Average Gyro Accel Y
Feat 4	19	37	Std Dev Gyro Accel Y
Feat 5	15	53	Average Gyro Accel Z
Feat 6	Removed by CM	Removed by CM	Std Dev Gyro Accel Z
Feat 7	**0**	**0**	**Average Gyro Ang Vel X**
Feat 8	2	28	Std Dev Gyro Ang Vel X
Feat 9	**0**	**0**	**Average Gyro Ang Vel Y**
Feat 10	Removed by CM	Removed by CM	Std Dev Gyro Ang Vel Y
Feat 11	**0**	**0**	**Average Gyro Ang Vel Z**
Feat 12	10	26	Std Dev Gyro Ang Vel Z
Feat 13	14	52	Average Accel X
Feat 14	Removed by CM	Removed by CM	Std Dev Accel X
Feat 15	15	58	Average Accel Y
Feat 16	21	37	Std Dev Accel Y
Feat 17	8	53	Average Accel Z
Feat 18	Removed by CM	Removed by CM	Std Dev Accel Z
Feat 19	0	18	Heart Rate
Feat 20	0	16	Skin Temperature
Feat 21	3	29	∆ Pedometer
Feat 22	13	29	∆ Distance
Feat 23	2	31	Speed
Feat 24	0	29	Pace
Feat 25	1	15	∆ Calories
Feat 26	**0**	**1**	**Uv**

Due to the lack of use it is feasible to eliminate these features from the PRIDE dataset without losing data representativeness. A special case is 26, which is never used in the analysis of all dimensions at a time, and it is only used 1 time in the analysis across all first five dimensions. For that reason, we decided to removed it from the dataset; its frequency contribution is too low to be consider relevant. Some features were removed before the PCA as a result of the CM analysis. Furthermore, features 19 and 20 are never used in the analysis of all dimensions at a time. However, their frequency of use in the analysis across all first five dimensions does not allow us to remove them from the PRIDE TD dataset since its frequency is as low as 26.

In total, 9 features were removed from the PRIDE TD dataset without losing data representativeness. The features are Std Dev Gyro Accel X, Std Dev Gyro Accel Z, Average Gyro Ang Vel X, Average Gyro Ang Vel Y, Std Dev Gyro Ang Vel Y, Average Gyro Ang Vel Z, Std Dev Accel X, Std Dev Accel Z, and Uv.

In Table 11 and Table 12 we shown the AUC obtained by OCKRA and ocSVM classifiers with the PRIDE TD dataset with all features and removing those that result from this analysis. Furthermore, in Table 13 we showed the comparison of the execution time of the previous classifiers for two users, the one with more data and the one with fewer data.

Table 8: Correlation matrix analysis over PRIDE dataset in Frequency-Domain

Feature number	Feature name	
Feat 1	23	Energy GyroSensor XAccel
Feat 3	23	Standard Deviation Energy GyroSensor XAccel
Feat 5	23	Peak DFT Bin GyroSensor XAccel
Feat 7	23	Peak Magnitude GyroSensor XAccel
Feat 10	23	Peak Energy GyroSensor XAccel
Feat 11	22	Energy GyroSensor YAccel
Feat 15	23	Peak DFT Bin GyroSensor YAccel
Feat 17	23	Peak Magnitude GyroSensor YAccel
Feat 18	22	Entropy GyroSensor YAccel
Feat 20	23	Peak Energy GyroSensor YAccel
Feat 21	23	Energy GyroSensor ZAccel

6 PCA and correlation analysis PRIDE subset on Frequency-Domain

For sake of simplicity, we only show those features that users agree that are highly-correlated.

The results in Table 8 show that all users report features 1, 3, 5, 7, 10, 15, 17, 20 and 21 as highly-correlated. Since these features are highly correlated (more than 0.75) with others, they were removed from the PRIDE FD dataset. Additionally, features 11 and 18 are reported by 22 users with high correlations with others features. Then, these two features were eliminated due to the number of users that report them. In total, 11 features were removed from the PRIDE FD dataset, reducing its features from 98 to 87.

Once removed the redundant features obtained by the CM, we performed the PCA. For sake of simplicity, we only show those features that are not present in all first 8 dimensions that retain the 60 % of data variability. The frequency of use in dimensions 1 to 8 is zero.

According with the results in Table 9, 23 features should be removed from the PRIDE FD dataset. However, we make a closer analysis to obtain the frequencies of use of each of these features per dimension across all users. Thus, we can remove features whose contributions to data variability are fewer.

The results in Table 10 show that features 38, 48, 58 and 98 are never used in PCA analysis; that is, these features are not contributing to explain the data variability. Due to the lack of use it is feasible to eliminate these features from the PRIDE dataset without losing data representativeness. In total, 15 features (11 from CM and 4 from PCA) were removed from the PRIDE FD dataset without losing data representativeness. In Table 11 and Table 12 we shown the AUC obtained by ocSVM and OCKRA classifiers with the PRIDE FD dataset with all features and removing those that result from this analysis. Furthermore, in Table 14 we showed the comparison of the execution time of the previous classifiers for two users, the one with more data and the one with fewer data.

Table 9: Principal component analysis over PRIDE dataset in Frequency-Domain – part 1

Feature number	Feature name
Feat 6	Spectral Entropy GyroSensor XAccel
Feat 9	Peak Frequency GyroSensor XAccel
Feat 16	Spectral Entropy GyroSensor YAccel
Feat 19	Peak Frequency GyroSensor YAccel
Feat 26	Spectral Entropy GyroSensor ZAccel
Feat 29	Peak Frequency GyroSensor ZAccel
Feat 36	Spectral Entropy GyroSensor XAngVel
Feat 38	Entropy GyroSensor XAngVel
Feat 39	Peak Frequency GyroSensor XAngVel
Feat 46	Spectral Entropy GyroSensor YAngVel
Feat 48	Entropy GyroSensor YAngVel
Feat 49	Peak Frequency GyroSensor YAngVel
Feat 56	Spectral Entropy GyroSensor ZAngVel
Feat 58	Entropy GyroSensor ZAngVel
Feat 59	Peak Frequency GyroSensor ZAngVel
Feat 66	Spectral Entropy AccelSensor X
Feat 69	Peak Frequency AccelSensor X
Feat 76	Spectral Entropy AccelSensor Y
Feat 79	Peak Frequency AccelSensor Y
Feat 89	Peak Frequency AccelSensor Z
Feat 91	Heart Rate
Feat 96	Pace
Feat 98	UV

Table 10: Principal component analysis over PRIDE dataset in Frequency-Domain – part 2

Feature number	Frequency of use across 8 dimensions	Feature name
Feat 6	32	Spectral Entropy GyroSensor XAccel
Feat 9	32	Peak Frequency GyroSensor XAccel
Feat 16	41	Spectral Entropy GyroSensor YAccel
Feat 19	31	Peak Frequency GyroSensor YAccel
Feat 26	40	Spectral Entropy GyroSensor ZAccel
Feat 29	35	Peak Frequency GyroSensor ZAccel
Feat 36	19	Spectral Entropy GyroSensor XAngVel
Feat 38	**0**	**Entropy GyroSensor XAngVel**
Feat 39	19	Peak Frequency GyroSensor XAngVel
Feat 46	27	Spectral Entropy GyroSensor YAngVel
Feat 48	**0**	**Entropy GyroSensor YAngVel**
Feat 49	20	Peak Frequency GyroSensor YAngVel
Feat 56	28	Spectral Entropy GyroSensor ZAngVel
Feat 58	**0**	**Entropy GyroSensor ZAngVel**
Feat 59	20	Peak Frequency GyroSensor ZAngVel
Feat 66	28	Spectral Entropy AccelSensor X
Feat 69	32	Peak Frequency AccelSensor X
Feat 76	38	Spectral Entropy AccelSensor Y
Feat 79	31	Peak Frequency AccelSensor Y
Feat 89	35	Peak Frequency AccelSensor Z
Feat 91	44	Heart Rate
Feat 96	13	Pace
Feat 98	**0**	**UV**

7 Partial Results and Discussion

Users want to be protected by an ideal classifier, which can correctly discriminate every possible abnormal behaviour from that of a normal user. Therefore, the aim is to build classifiers that maximize true positive classifications (i.e. true abnormal conditions) while minimizing false positive ones (i.e. false abnormal or dangerous situations). Therefore, the classifiers were evaluated using the following performance indicators.

- **AUC**: The AUC of the TPR versus the false positive detection rate (FPR), which indicates the general performance of the classifier for all FPR rates.

The following classifiers were compared for both time-domain and frequency-domain.

- **ocSVM**: The implementation of ocSVM [10] included in LibSVM [2] with the default parameter values ($\gamma = 0.038$ and $\nu = 0.5$) and using the radial basis function kernel.
- **Parzen**: Parzen window classifier using the Euclidean distance [3]. For every training dataset, the classifier computes the width of the Parzen-window by averaging the distances between objects sampled every 60 s.[3]
- **k-means1**: A version of the Parzen window classifier based on k-means [9]. k-means1 classifies new objects based only on the closest centre of the cluster.
- **k-means2**: A version of the Parzen window classifier based on k-means [4]. k-means2 classifies new objects using all the centres of the clusters.

We divided our experiments using the following TD and FD sets and subsets, as results of PCA a correlations analysis:

1. All TD features = PRIDE with 26 features
2. TD features subset = PRIDE with 19 features
3. All FD features = PRIDE with 98 features
4. FD features subset = PRIDE with 83 features

Next, we present our partial results.

In order to quantify the differences among the algorithms, the average of AUC results was computed for all the test subjects.

[3] This procedure saved approximately 7 days when computing the distances per test subject using an Intel Core i7-4600M CPU at 2.90 GHz.

Table 11: Area (percentage) under the curve for TPR versus FPR for ocSVM classifier

Test Subject	TD all features	TD features subset	FD all features	FD features subset
TS 1	97.3	97.3	79.1	78.5
TS 2	94.5	94.3	82.2	81.6
TS 3	87.4	87.2	74.5	73.9
TS 4	83.9	82.1	57.7	57.1
TS 5	80.8	80.8	65.7	65.8
TS 6	96.1	96.1	81.8	81.8
TS 7	69.4	68.1	64.9	64.1
TS 8	93.8	94.0	73.2	71.6
TS 9	95.3	95.5	76.4	75.6
TS 10	94.0	94.3	70.0	69.8
TS 11	93.4	93.8	66.5	66.1
TS 12	74.6	73.4	73.2	73.3
TS 13	75.8	73.4	74.1	73.4
TS 14	78.0	78.2	63.0	62.9
TS 15	93.8	94.4	71.5	70.8
TS 16	83.2	83.0	73.6	73.2
TS 17	98.1	99.0	82.5	82.1
TS 18	89.1	89.0	77.0	77.0
TS 19	89.4	90.0	64.7	64.2
TS 20	90.5	90.2	78.0	77.8
TS 21	98.4	98.4	89.5	89.4
TS 22	78.3	77.8	70.8	70.1
TS 23	53.0	52.6	63.3	62.9
Average	**86.44**	**86.2**	**72.8**	**72.3**

7.1 PRIDE TD dataset

From Table 13, we can notice that the time required to train a ocSVM classifier with all features of the PRIDE TD dataset lies between one hour to 10 hours, while training time required for a OCKRA classifier with a subset of features goes from 45 minutes to 11 hours. It is noteworthy that ocSVM classifiers were trained with a sampled dataset, for all features and a subset of them, of 1/10 instances. Furthermore, for OCKRA classifiers we can observe that training time varies from 25 minutes to 5:30 hours and from one minute to three minutes for the PRIDE TD dataset with all features and PRIDE TD dataset with a subset of features respectively. It is noteworthy that OCKRA classifiers were trained with a sampled dataset, for all features and a subset of them, of 1/10 instances. Training times of both classifiers were reduced by training them with a subset of features and sampled instances. Training time reduction is more notable in OCKRA classifiers, reducing it from half an hour to a couple of minutes or seconds.

7.2 PRIDE FD dataset

From Table 14, we can pinpoint that the time required to train a ocSVM classifier with all features of the PRIDE FD dataset varies from one hour to 14 hours for the 23 current users. On the other hand, the time required to train a ocSVM classifier with the subset of features previously obtained for the PRIDE FD dataset varies from one

Table 12: Area (percentage) under the curve for TPR versus FPR for OCKRA classifier

Test Subject	TD all features	TD features subset	FD all features	FD features subset
TS 1	98.8	95.5	78.0	81.0
TS 2	95.7	92.0	85.5	82.9
TS 3	91.2	84.1	82.4	81.9
TS 4	88.2	83.6	61.4	61.0
TS 5	90.2	71.5	68.8	61.9
TS 6	98.2	97.4	87.9	82.3
TS 7	79.2	76.9	65.6	59.5
TS 8	92.4	86.8	77.6	72.8
TS 9	92.7	89.3	81.5	77.8
TS 10	93.7	91.5	69.3	71.9
TS 11	90.9	79.4	69.9	66.8
TS 12	80.3	77.6	71.4	69.4
TS 13	80.5	76.0	70.7	69.2
TS 14	81.9	79.0	66.8	65.1
TS 15	94.5	89.9	77.4	69.3
TS 16	87.9	84.3	73.0	73.8
TS 17	98.0	84.1	81.6	81.9
TS 18	86.9	75.9	70.6	72.5
TS 19	89.6	86.3	64.5	61.8
TS 20	92.2	88.2	79.8	73.6
TS 21	97.9	94.2	87.2	88.3
TS 22	79.2	77.4	70.2	71.5
TS 23	68.9	64.2	72.3	60.9
Average	**89.1**	**83.7**	**74.5**	**72.0**

hour to 13 hours for the same 23 users. It is noteworthy that ocSVM classifiers were trained with a sampled dataset, for all features and a subset of them, of 1/10 instances. If we apply the sampling approach followed in the ocSVM classifier, then the training time goes from three minutes to 18 minutes. In contrast to the same PRIDE FD dataset with a subset of features but with the proposed sampling approach, the training time took between 2:21 minutes to 16 minutes. For both classifiers (ocSVM and OCKRA) the time needed to train each was reduced by using a subset of features and a sampling method of 1/10 instances. While training time for ocSVM classifiers has no significant difference between using a subset of features or all the features, there was a training time reduction from hours to minutes for OCKRA classifiers.

Table 13: One-class classifiers runtime comparison in Time-Domain

Classifier	Features	User	Total training time in hh:mm:ss format
ocSVM	All	1	09:44:11
ocSVM	Subset	1	10:50:48
ocSVM	All	17	01:04:05
ocSVM	Subset	17	00:44:50
OCKRA	All	1	05:29:26
OCKRA	Subset	1	00:02:41
OCKRA	All	17	00:25:06
OCKRA	Subset	17	00:00:45

Table 14: One-class classifiers runtime comparison in Frequency-Domain

Classifier	Features	User	Total training time in hh:mm:ss format
ocSVM	All	1	13:55:27
ocSVM	Subset	1	12:49:15
ocSVM	All	17	01:07:16
ocSVM	Subset	17	01:04:41
OCKRA	All	1	00:17:39
OCKRA	Subset	1	00:15:52
OCKRA	All	17	00:02:47
OCKRA	Subset	17	00:02:21

Reduction of processing time is crucial when developing an on-line monitoring system for personal risk detection. The experiments done for visualization show a better manner to determine whenever a person is in risk, allowing a decision maker to make an informed decision based on the visual aids. For the conversion of the features in PRIDE to the frequency-domain, we are still working on this process so our results are only partial.

7.3 CPU and Memory Costs of Experiments

All experiments were performed using a 64-core cluster with 128 GB from HPI Lab and 32 cores from ITESM CEM. Approximately, a total of 1 366 200 hours (56 925 days) were required to perform the visualization experiments; where CPU hours = 10 days (time to train and test the one-class classifiers) ×49 values of k in k-means algorithm ×5 folds ×23 users + 25 days (time to evaluate the one-class classifiers) ×23 users. Calculations to transform the PRIDE dataset to time and frequency-domain approximately took 184 hours (8 days); where CPU hours = 7 minutes (to obtain time and frequency-domain features) ×1 user day log ×7 user's day logs ×23 users.

References

[1] A. Y. Barrera-Animas, L. A. Trejo, M. A. Medina-Pérez, R. Monroy, J. B. Camiña, and F. Godínez. "Online Personal Risk Detection Based on Behavioural and Physiological Patterns". In: *Information Sciences* 384 (Apr. 2017), pages 281–297. DOI: doi:10.1016/j.ins.2016.08.006.

[2] C.-C. Chang and C.-J. Lin. "LIBSVM: A Library for Support Vector Machines". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 2.3 (2011), pages 1–27. DOI: 10.1145/1961189.1961199.

[3] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. 2nd. Wiley-Interscience, 2001. 680 pages.

[4] G. Giacinto, R. Perdisci, M. D. Rio, and F. Roli. "Intrusion Detection in Computer Networks by a Modular Ensemble of One-Class Classifiers". In: *Informa-*

tion Fusion 9 (1 Jan. 2008). Special Issue on Applications of Ensemble Methods, pages 69–82. DOI: 10.1016/j.inffus.2006.10.002.

[5] A. López-Cuevas, M. A. Medina-Pérez, R. Monroy, J. Ramírez-Márquez, and L. A. Trejo. "FiToViz: A Visualisation Approach for Real-time Risk Situation Awareness - (under revision)". In: *IEEE* (2017).

[6] A. Maxhuni, A. Muñoz-Meléndez, V. Osmani, H. Perez, O. Mayora, and E. F. Morales. "Classification of bipolar disorder episodes based on analysis of voice and motor activity of patients". In: *Pervasive and Mobile Computing* 31 (2016), pages 50–66.

[7] M. A. Medina-Pérez, R. Monroy, J. B. Camiña, and M. García-Borroto. "Bagging-TPMiner: a classifier ensemble for masquerader detection based on typical objects". In: *Soft Computing* 21.3 (2017), pages 557–569. ISSN: 1433-7479. DOI: 10.1007/s00500-016-2278-8.

[8] J. Rodríguez, A. Y. Barrera-Animas, L. A. Trejo, M. A. Medina-Pérez, and R. Monroy. "Ensemble of One-Class Classifiers for Personal Risk Detection Based on Wearable Sensor Data". In: *Sensors* 16.10 (2016), page 1619.

[9] D. M. J. Tax and R. P. W. Duin. "Combining One-Class Classifiers. Second International Workshop, MCS 2001". In: *Multiple Classifier Systems*. Volume 2096. Springer Berlin Heidelberg, July 2001, pages 299–308. DOI: 10.1007/3-540-48219-9_30.

[10] V. N. Vapnik. *Statistical Learning Theory*. 1st. Volume 1. Wiley-Interscience, 1998. 768 pages.

A Big Data Science Experiment
Protecting Minors on Social Media Platforms

Estée van der Walt and Jan H. P. Eloff

Department of Computer Science
Security & Data Science Research Group
University of Pretoria, South Africa
estee.vanderwalt@gmail.com,eloff@cs.up.ac.za

Individuals are exposed to many threats on social media platforms. These threats can originate from non-human or human accounts. Past research work focused on the detection of non-human accounts as opposed to finding human accounts posing a threat on social media platforms. For the research at hand, we focus on humans that can harm other individuals by hiding their identity. An example of such cases involves the grooming of minors. Various supervised machine learning experiments are performed with the intention to not only detect such identity deception and protect minors, but also to identify those attributes or engineered features that contribute most towards the detection of identity deception. A final Identity Deception Score (IDS) is proposed with which these malicious individuals can be identified and further investigated by law enforcement.

1 Project idea

Data is set to grow to 44 zettabytes by 2020 [13]. This can largely be attributed to what is known as "big data" [6]. "Big data" shows characteristics, known as the 3Vs, of high volume, velocity, and variety [11]. Social media and the Internet of Things (IOT) are examples of such big data platforms.

Various cyber threats can be found on social media platforms (SMPs). The nature of SMPs, being a big data platform, makes it very difficult to detect these cyber threats. Identity deception are but one such example and the focus of the current research at hand. With identity deception, the presented or perceived identity of an entity is different from what is expected. A few examples of use cases of the outcomes of identity deception within SMPs are as follow:

- Influencing outcomes or results, like political campaigns [5].
- Enhancing or damaging the image of a company's brand [7, 8].
- Spreading fake news [3, 7].
- Pedophiles who lie about who they are to approach a minor [2].
- Grooming of individuals for some malicious purpose [9]

- Extremism recruitment [10]

It was found that these cyber threats originate from either non-human or human accounts. The non-human accounts are also known as bots [14]. Much research in the past has focused on the detection of these fake accounts generated by bots [1, 4, 12]. These fake accounts were identified either by their identity (attributes defining who they are), their behavior (their relationships), or the content that was posted through these accounts. The authors believe that the detection of fake accounts generated by humans should receive the same attention.

Current research towards the detection of identity deception by humans has been found lacking for the following reasons:

- The attributes available on social media platforms are found to be lacking deceptive identification information for humans.
- Much of the research looks at detecting deception via content instead of identity. Looking at content could be very resource intensive and the threat would probably be detected too late.
- It is difficult to explain the results from models trained by machine learning to detect deception.

The current research project proposes to address these issues. The research at hand proposes to identify new features to detect identity deception from humans but also use this knowledge to create an Identity Deception Score (IDS). The IDS will be intuitive and take account the fact that certain attributes or features contribute to identity deception to a different extent.

The research project has been divided into various methods discussed in more detail during previous research papers [15] and follows a scientific approach. The focus of this phase of the research, highlighted in figure 1, was to use the identified deceptive attributes from the previous phase towards identifying deception and apply them to various supervised machine learning experiments. The results are discussed in section 3 of this report.

Main deliverables

The main deliverables of the past six months were:

- To determine a ground truth corpus for input to supervised machine learning and the continuation of the research
- To analyze the effect of the skewness of data and determine the best method to cater for these scenarios
- To introduce new engineered features to the corpus in the hope to improve on previous results
- To evaluate these results

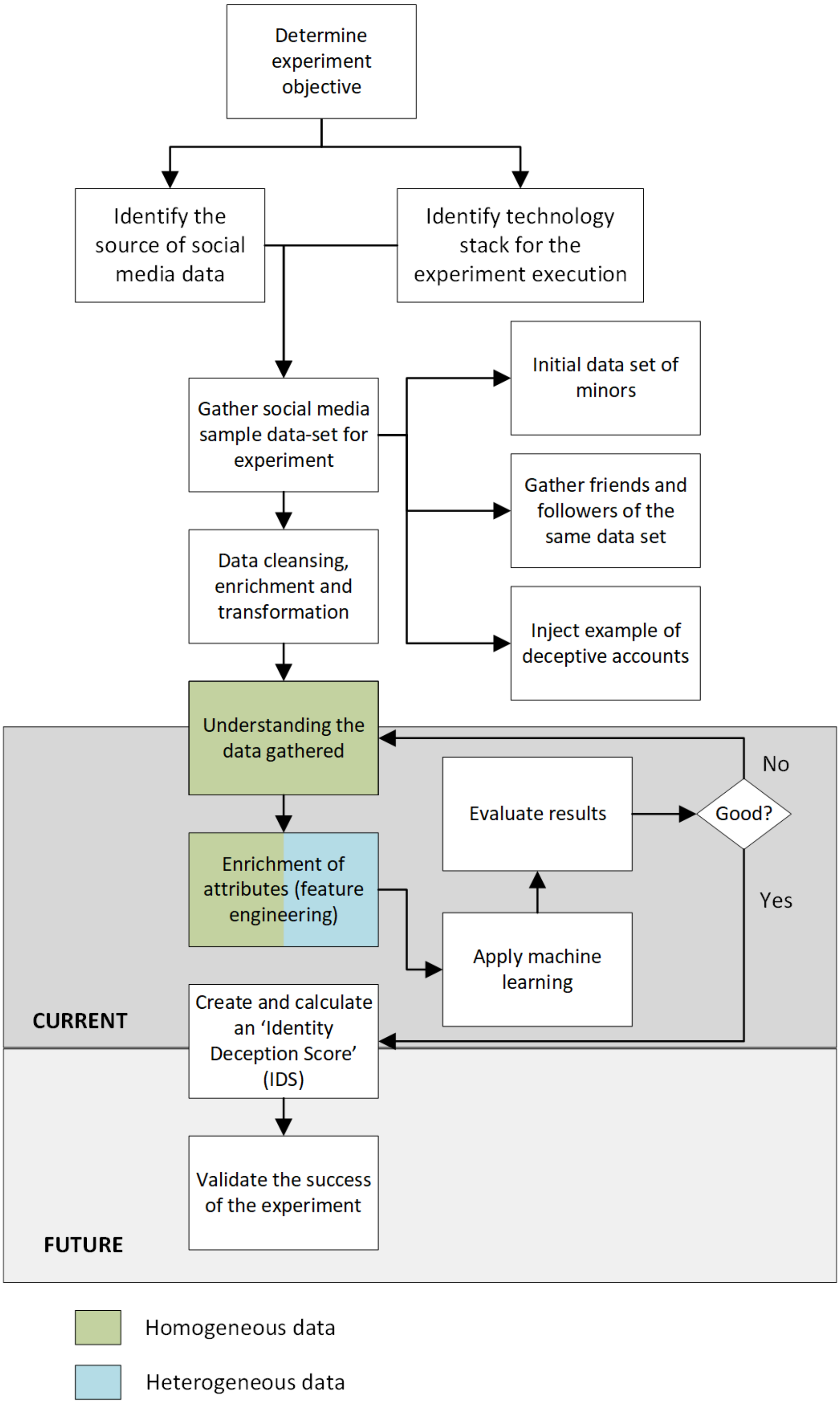

Determine experiment objective
Identify the source of social media data
Identify technology stack for the experiment execution
Gather social media sample data-set for experiment
Initial data set of minors
Gather friends and followers of the same data set
Inject example of deceptive accounts
Data cleansing, enrichment and transformation
Understanding the data gathered
Evaluate results
Good?
No
Yes
Enrichment of attributes (feature engineering)
Apply machine learning
CURRENT
Create and calculate an 'Identity Deception Score' (IDS)
Validate the success of the experiment
FUTURE
🟩 Homogeneous data
🟦 Heterogeneous data

Figure 1: The project process diagram

- To improve the results by understanding the weight certain features carry towards the detection of identity deception

2 Use of HPI Future SOC Lab resources

To reiterate past feedback, the following resources were used for the research at the HPI Future SOC lab:

- Twitter: The Twitter4j Java API was used to dump the data needed for the experiment in a big data repository.
- Hortonworks Hadoop 2.4: For the purposes of this experiment HDP Hadoop runs on an Ubuntu Linux virtual machine hosted in "The HPI Future SOC"-research lab in Potsdam, Germany. This machine contains 4 TB of storage, 8 GB RAM, 4 ×Intel Xeon CPU E5-2620 @2 GHz and 2 cores per CPU. Hadoop is well known for handling heterogeneous data in a low-cost distributed environment, which is a requirement for the experiment at hand.

Flume: Flume is used as one of the services offered in Hadoop to stream initial Twitter data into Hadoop and into SAP HANA.

Ambari: For administration of the Hadoop instance and starting/stopping the services like Flume.

- Java: Java is used to enrich the Twitter stream with additional information required for the experiment at hand and automate the data gathering process.
- SAP HANA: A SAP HANA instance is used which is hosted in "The HPI Future SOC"- research lab in Potsdam, Germany on a SUSE Linux operating system. The machine contains 4 TB of storage, 2 TB of RAM (1.4 TB effective) and 32 CPUs / 100 cores. The in-memory high-performance processing capabilities of SAP HANA enables almost instantaneous results for analytics.

The XS Engine from SAP HANA is used to accept streamed Tweets and populate the appropriate database tables.

- Machine learning APIs: Various tools are considered to perform classification, analysis and apply deep learning techniques on the data. These include the PAL library from SAP HANA, SciPy libraries in Python, Spark Mlib on Hadoop and the Hadoop Mahout service. For the research, R was the final choice. This decision was made due to support on this platform and libraries freely being available on the web community at a large scale.
- An additional Linux machine was provided for the lab to aid in the running of the CPU and memory intensive machine learning algorithms. The VM has 8 cores and 64 GB of RAM.

- Visualization of the results will be performed by the libraries in R and PowerBI where appropriate.

The following ancillary tools were used as part of the experiment:

- For connection to the FSOC lab we used the OpenVPN GUI as suggested by the lab.
- For connecting and configuration of the Linux VM instance we used Putty and WinSCP
- For connecting to the SAP HANA instance, we used SAP HANA Studio (Eclipse) 1.80.3

3 Findings in the Spring 2017 semester

Past research built a novel identity deception score (IDS) to understand its viability. The IDS was found to indicate deception, but further analysis was required with labeled data as it was impossible to know whether this score was accurate or not. The purpose of this phase of the research project was to combine data from known deceptive accounts with the mined corpus. This allowed for a new labelled corpus consisting of two classes of data; those that come from known deceptive accounts and those accounts which are not deceptive. This same set-up is typically seen in simple binary classification problems for which supervised machine learning is a solution. For this research, supervised machine learning used the corpus as input to train models to predict the expected outcome with the best possible accuracy; in this case whether an account is deceptive or not.

The final labelled corpus consisted of 154,417 accounts. Of these, 1000 were labelled as deceptive and the rest as not. Supervised machine learning experiments were performed initially to build a model to detect identity deception with the attributes found on social media platforms only. Eight machine learning algorithms were applied to the experiment which made use of 10-fold, 3 repeat cross validation. Synthetic Minority Over-sampling Technique (SMOTE) was used to cater for the skewness in data. Without SMOTE the trained models were bias towards the non-deceptive accounts.

Next, additional experiments were executed with the intention to improve on the first. The results from three experiments performed during the past semester will be discussed next.

3.1 First experiment

The first experiment made use of the attributes found on social media platforms alone to predict identity deception. The results from eight machine learning models are presented in Table 1a.

(**a**) Results from experiment 1

Model	**F1 Score**	**PR-AUC**
svmRadial	20.29 %	25.60 %
rf	54.66 %	68.22 %
J48	12.00 %	10.44 %
bayesglm	1.39 %	0.75 %
knn	16.93 %	35.19 %
Adaboost	19.75 %	68.35 %
rpart	21.40 %	27.72 %
nnet	5.95 %	6.18 %

(**b**) Results from experiment 2

Model	**F1 Score**	**PR-AUC**
svmRadial	7.49 %	8.05 %
rf	20.60 %	40.65 %
J48	19.43 %	12.50 %
bayesglm	4.16 %	2.23 %
knn	6.60 %	7.27 %
Adaboost	22.39 %	40.12 %
rpart	12.78 %	8.27 %
nnet	12.49 %	14.51 %

The rf model performed the best with an accuracy of 54.66 %. Figure 1, showing the Precision-Recall Area Under the Curve (PR-AUC) for all models, affirms these results. The PR-AUC shows the tradeoff between precision and recall for each model. A perfect model would have its precision at 1.00 consistently.

Lastly, entropy showed us that the friends count, profile image, UTC offset and language played an important role in the performance of the models.

3.2 Second experiment

Previous research on detecting non-human accounts generated engineered features for this specific purpose. These same engineered features were applied during the second experiment with the intention to understand if these features would improve the accuracy of the models trained during the first experiment. This would also indicate whether features used to detect non-human accounts can be applied to human accounts as well. The results from eight machine learning models are presented in Table 1b.

The Adaboost model performed the best with an accuracy of 22.39 %. This result is not very good as it is well below predicting the correct outcome by chance or 50 %. Figure 2, showing the Precision-Recall Area Under the Curve (PR-AUC) for all models, affirms these results. Most models are below 0.50 precision.

Regardless of the result, entropy still showed that the length of the profile name and the fact that the account is geo enabled, played an important role in the performance of the models.

3.3 Third experiment

The last experiment looks towards previous research in the social sciences. Psychology has shown why people lie. New features were engineered given this knowledge. The results from eight machine learning models are presented in Table 2.

The rpart model performed the best with an accuracy of 79.09 %. This showed good improvement on both the results from the first two experiments. Figure 3, showing

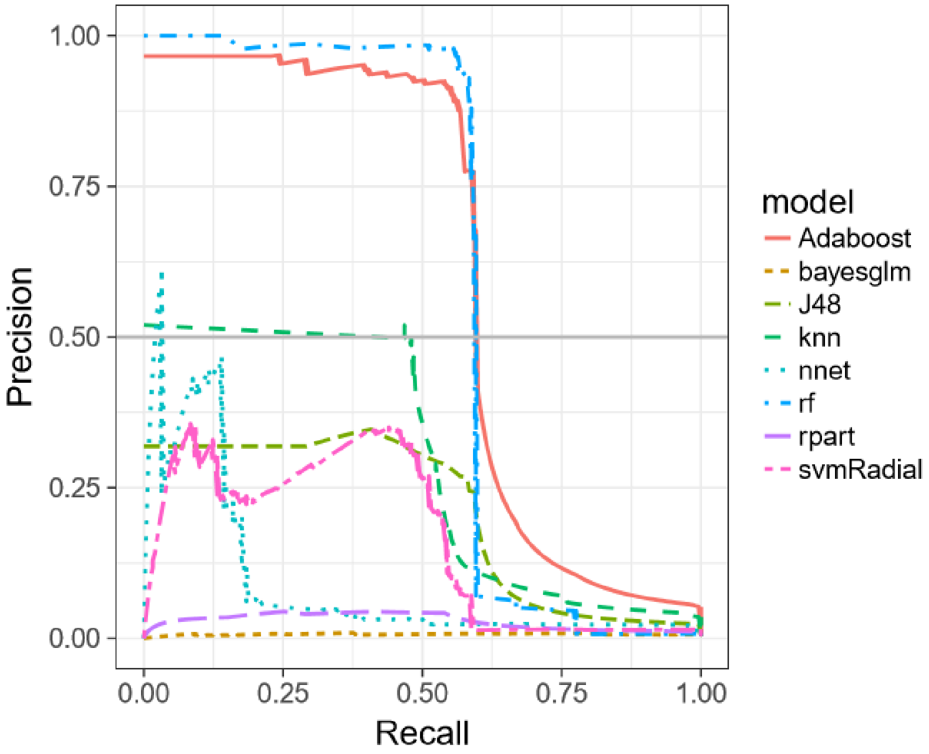


Figure 2: PR-AUC for experiment 1

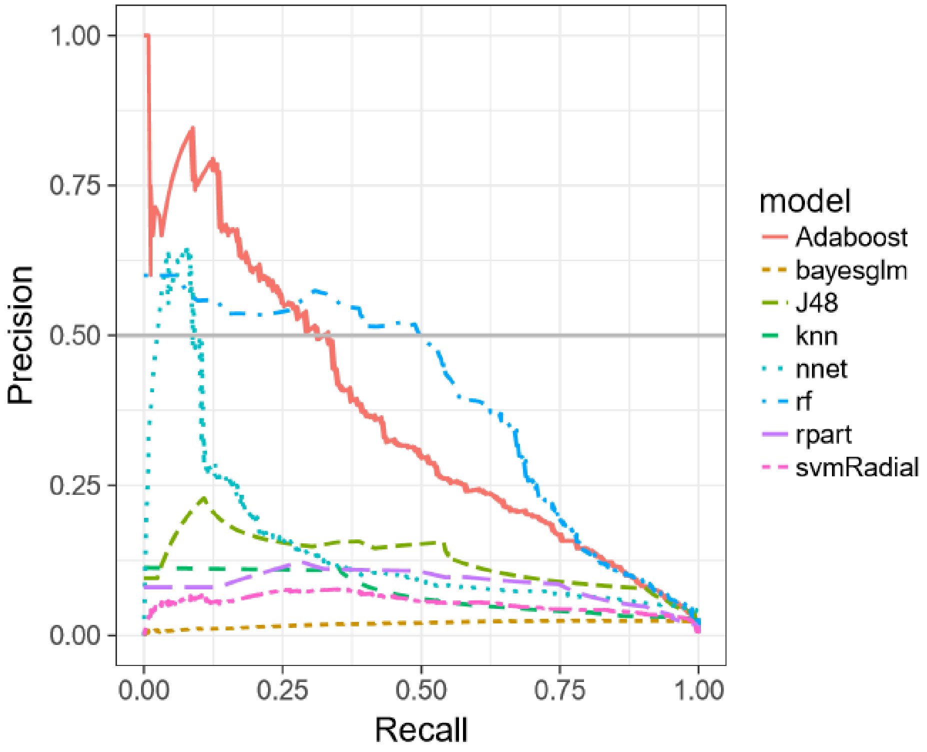


Figure 3: PR-AUC for experiment 2

Table 2: Results from experiment 3

Model	F1 Score	PR-AUC
svmRadial	10.29 %	72.46 %
rf	10.91 %	76.49 %
J48	10.93 %	66.19 %
bayesglm	14.33 %	72.46 %
knn	9.51 %	72.66 %
Adaboost	10.97 %	5.62 %
rpart	79.07 %	63.39 %
nnet	11.09 %	74.89 %

the Precision-Recall Area Under the Curve (PR-AUC) for all models, affirms these results. Most models are above 0.50 precision.

The entropy for this experiment showed that location was a very important role player towards the accuracy of the models.

4 Architecture

The SAP HANA instance, virtual machines and storage was provided by the HPI FSOC research lab and the following is worth mentioning:

- There were no issues in connection.
- The lab was always responsive and helpful in handling any queries.
- The environment is very powerful, and more than enough resources are available which makes the HPI FSOC research lab facilities ideal for the experiment at hand.
- Without the additional VM with more cores, we would not have been able to perform the machine learning computations.

Overall, we found that the environment and its power enabled the collection and handling of a big dataset without issue. The support of the HPI FSOC research lab is greatly appreciated.

5 Next steps for 2017/2018

The deliverables for this next phase are as follow:

- To combine all results from previous experiments.

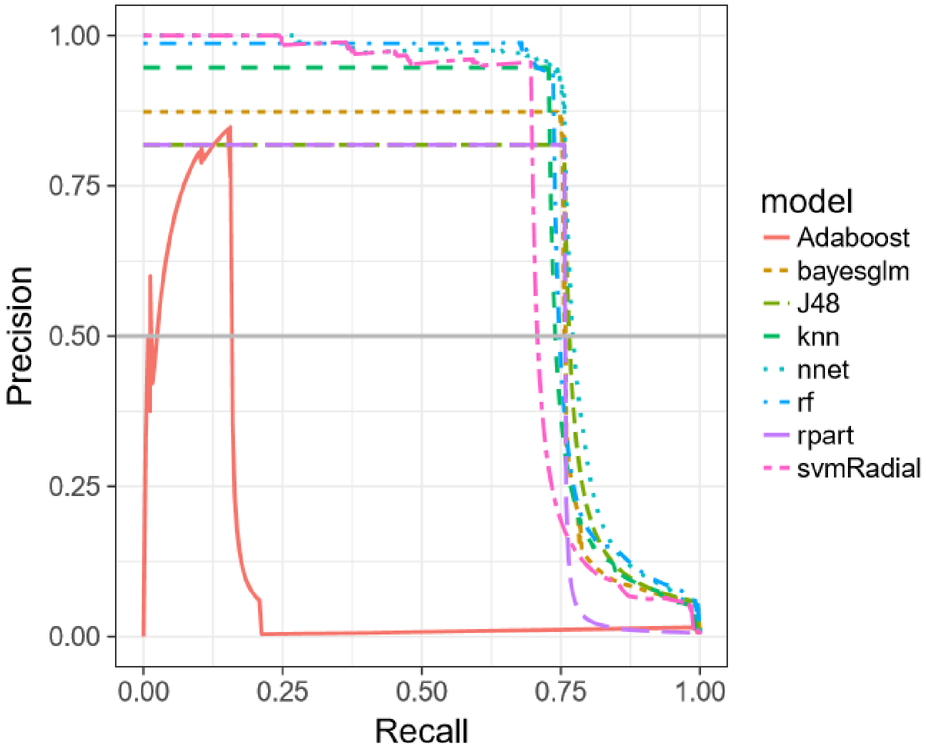


Figure 4: PR-AUC for experiment 3

- To compare the performance of all machine learning algorithms used for all experiments.

- The problem with machine learning models are that they are seldom transparent. Next, the input from all experiments will be applied to build an IDS per account. The IDS is proposed to be the result from a simpler model, still able to detect identity deception, but intuitive and explainable.

- Lastly, the IDS produced per account will be evaluated to discern the overall performance of this model.

- A conclusion and proposal for future work from the research will be presented.

References

[1] B. van den Belt. *How to recognize Twitter bots: 7 signals to look out for*. Aug. 20, 2012. URL: https://www.stateofdigital.com/how-to-recognize-twitter-bots-6-signals-to-look-out-for/.

[2] D. Bogdanova, P. Rosso, and T. Solorio. "Exploring high-level features for detecting cyberpedophilia". In: *Computer Speech & Language* 28 (2014), pages 108–120. DOI: 10.1016/j.csl.2013.04.007.

[3] C. Chen, K. Wu, V. Srinivasan, and X. Zhang. "Battling the internet water army: Detection of hidden paid posters". In: *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 2013, pages 116–120. DOI: 10.1145/2492517.2492637.

[4] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia. "Who is tweeting on Twitter: human, bot, or cyborg?" In: *Proceedings of the 26th annual computer security applications conference*. 2010, pages 21–30. DOI: 10.1145/1920261.1920265.

[5] N. J. Conroy, V. L. Rubin, and Y. Chen. *Automatic Deception Detection: Methods for Finding Fake*. 2015.

[6] F. X. Diebold. *A Personal Perspective on the Origin(s) and Development of "Big Data": The Phenomenon, the Term, and the Discipline*. 2012. DOI: 10.2139/ssrn.2202843.

[7] B. Drasch, J. Huber, S. Panz, and F. Probst. *Detecting Online Firestorms in Social Media*. 2015.

[8] S. Gurajala, J. S. White, B. Hudson, and J. N. Matthews. "Fake Twitter accounts: Profile characteristics obtained using an activity-based pattern detection approach". In: *Proceedings of the 2015 International Conference on Social Media & Society*. 2015, page 9. DOI: 10.1145/2789187.2789206.

[9] S. Kierkegaard. "Cybering: online grooming and ageplay". In: *Computer Law & Security Review* 24 (2008), pages 41–55. DOI: 10.1016/j.clsr.2007.11.004.

[10] B. I. Koerner. "Why ISIS is winning the social media war". In: *Wired* (2016).

[11] D. Laney. *3D Data Management: Controlling data volume*. 2001.

[12] V. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, F. Menczer, A. Stevens, A. Dekhtyar, S. Gao, T. Hogg, F. Kooti, Y. Liu, O. Varol, P. Shiralkar, V. Vydiswaran, Q. Mei, and T. Hwang. "The DARPA Twitter Bot Challenge". In: 49.6 (2016), pages 38–46. DOI: 10.1109/MC.2016.183. arXiv: 1601.05140 [cs.SI].

[13] V. Turner, J. F. Gantz, D. Reinsel, and S. Minton. *The digital universe of opportunities: Rich data and the increasing value of the internet of things*. IDC Analyze the Future, 2014.

[14] Unknown. *How to spot fake Twitter followers (bots)?* 2015.

[15] E. V. der Walt and J. H. P. Eloff. "Protecting minors on social media platforms - A Big Data Science experiment". presented at the HPI Cloud Symposium "Operating the Cloud", Potsdam, Germany. 2015.

Analysis of Textual Stance Detection for Visual Integration

Thilini Cooray

Singapore University of Technology and Design
muthuthanthrige@mymail.sutd.edg.sg

Stance detection has recently drawn a huge interest in Natural Language Processing community due to the variety of applications it can support. Social media networks provide a great platform for people around the world to state their stance on vast number of different topics. This study focussed on analysing current state of stance detection on social media data, experimenting on novel approaches to evaluate their capabilities of addressing issues on current methods and discuss possibilities of integrating visual information to further enhance existing approaches.

1 Introduction

Many people tend to express their opinions through online blogs, forums and social media platforms more and more and the World Wide Web (WWW) has become their audience. Whether you want to know about a household appliance, a restaurant, an electronic device or a degree program, WWW is the first place you refer. A common nature of many of these online posts is that, they are always followed by supporting and counter arguments for them. Different people from around the world representing different ethnic groups, different age groups and different professions express their ideas. These data are a gold mine for many companies due to the variety of applications which can nurture from them. For an example, analysing these posts and understanding public opinion of their products help companies to adjust their marketing strategies. Also this data can be vital for election candidates or governments to evaluate public opinion on themselves. Therefore, it is essential for research communities to work on approaches which can accurately discover hidden meaning of these data. Stance detection is a task which was introduced to classify whether the author of a text is in agreement, disagreement or expressing neither of them about a given target. Text can be a news article, a blog, a tweet or a Facebook post and the target can be a person, organization, movement, product, news headline etc. Stance detection is a typical task which can help any entity such as companies and election candidates to accurately understand public's position about them and their rivals.

Here is a more general task definition was formulated based on the definition provided by [12]: Given a corpus of documents (tweets, online news comments, news articles, blog posts etc.), and a target entity (person, organization, movement, policy etc.), automatic natural language processing systems should be able to determine whether the author of the article is in favour of the given target, against it or neither can be inferred.

Formally this can be written as, for a given set of documents D related to a target T, the goal of stance detection is to retrieve the mapping $s_T : D \rightarrow \{favor, against, neither\}$ for any element $d \in D$. Stance or the position of the author towards a target can be expressed in several different ways. Following are some examples. These examples are used from the training dataset of tweets presented in SemEval-2016 Task 6: Detecting Stance in Tweets [9].

Direct expressions Author directly mentions the target and his/her position about the target.

Document: Hillary is our best choice if we truly want to continue being a progressive nation. #Ohio
Target: Hillary Clinton
Stance: Favour

Direct expressions with different textual representations Author provides his/her position directly about the target, however the wordings he uses are not exactly similar. In the following example, a human can directly understand that pro-life support is against legalization of abortion.

Document: I will agree to gay marriage if you agree to pro-life and then adopt babies that would have been aborted.
Target: Legalization of Abortion
Stance: Against

Indirect expressions Author provides his position towards a different entity/fact which is related to the target. In the following example, the author is talking about emission of CO_2 as a bad thing. Emission of CO_2 badly effects climate change. Humans can deduce that the author is agreeing that climate change is a concern with that contextual understanding of the relatedness of those facts.

Document: Every human commits original sin with it's first out-breath of CO2. Mankind is fallen. #bible #SemST
Target: Climate Change is a Real Concern
Stance: Favour

The style authors used to express their stance largely differs with the medium of communication. Above examples were expressed in Tweets where authors simply express their opinion about different topics unrelated to each other. On the other hand, there are online debates where people provide their position about target entities of the debate. People tend to talk about both sides in their post with their position about each, they also do not mention the topic explicitly in the post and mostly express themselves as counter arguments to someone else's post [13]. Also there can be newspaper articles where the journalists keep on elaborating on misdeeds of a political party or a politicians without directly mentioning them by which they support the opposition.

2 Related Work

Early work on Stance Detection was aiming at sources like congressional debates [14] and online forum debates [13, 17] where a group of people discuss about a given topic and researchers try to automatically extract whether speakers are supporting the topic or not. All of these works have mainly focussed on extracting relationships among discourse segments by studying the discussion structure and land on stance classification based on those observations.

With the tide of social media, many users tend to express their opinions about different topics in their social media profiles without getting limited to online forums/blogs. Therefore it is inevitable for social media to form discussions with active and passive participation of millions of users on different topics. [11] consider this new trend and form a framework for stance classification in Twitter. Authors have mentioned huge number of posts, usage of informal language such as slang, abbreviations and emoticons and the word limit to 140 as some of the challenges in analysing social media text compared to others. Their framework is built upon a label propagation mechanism whether they first manually label stance for a limited number of tweets posted by seed users such as politicians. Then they create a bit bigger set of stance labelled tweets using retweeting nature in Twitter. Based on the assumption that if a user retweets two tweets within a short amount of time only if both tweets have similar opinion towards a target, they form a retweet co-occurrence matrix and finally retrieve stance labels using that. Once they created their training set, they have used a supervised approach for stance detection.

Above work had given rise to several other follow up work of stance detection from social media. SemEval-2016 Task 6 on detecting stance in Tweets [9] can be identified as one of the more recent such work. However this task aimed at detecting stance of individual tweets on a given topic regardless of conversational flow it took part in. Organizers had given a labelled dataset of tweets for five targets (Hillary Clinton, Climate Change is a Real Concern, Legalization of Abortion, Atheism and Feminist Movement) both supporting and disagreeing with the target. Competition had two tasks where the first one was to train a supervised classifier using given data and the second task is to detect stance for an unlabelled tweet set whose target is Donald Trump. 19 teams competed for the supervised task. There was a vast diversity in approaches these teams used regarding features, models and data sources they used. Some approaches [6, 8, 19] have used traditional linguistic features such as word n-grams, character n-grams, TF-IDF, dependency and Part-of-Speech tags along with sentiment lexicons while others [1, 21] have used word embeddings pre-trained on a large corpus such as Google News to represent tweets. When it comes to models, several teams have used neural models such as RNN [21], CNN [4, 16] while others relied on classifiers such as SVM [10], Naive Bayes [1] and ensemble methods [7, 15]. Some teams have also used external noisy data to support their solutions such as the winning system by MITRE [21] who used a large unlabelled tweet set to learn word embeddings to represent tweets. These approaches performed well for classifying separate target with a separate classifier, however they were unable to beat the SVM classifier with word and character n-grams when classifying all targets together. Another significant observation was that these classifiers performed very poorly when the target of opinion in the tweet is not the target we are interested in. 9 teams participated for the second task which is a weakly supervised task. Some teams [18] have used noisy data and rule based methods to while some [22] have tried out methods to generalize supervised dataset to address this.

3 Methodology

All these existing work has considered Stance Detection task as a classification problem. We understood that there is a natural tendency of a ranking nature in this problem. Because these texts are not always stating only about a single target. They mostly tend to talk about different targets as a way to support their stance towards an entity. And also people tend to discuss both pro and con of a given target before stating their final opinion. Therefore we identified that it is better to model this task as a ranking problem and experiment on it to check whether it can surpass current classification base methods.

3.1 Learning to Rank

This is a class of machine learning techniques which are used for solving ranking problems. Unlike classification or regression problems where the algorithm aims at providing a class label or a value to each single instance, ranking algorithms look for a way to output the optimal ordering of elements in their list. As mentioned by [20], following are some significant differences among classification and ranking in depth.

Data set classification dataset for a supervised problem consists of entries $D = (x_1, y_1), ..., (x_n, y_n)$ where input document x_1 is represented as a feature vector and is associated with a class label y_i. Ranking dataset consists again with entries $D = (x_1, y_1), ..., (x_n, y_n)$ where input document is the same feature vector as classification, however the output is no longer a distinct label independent of other entries in the dataset. Labels define an ordering in a way y_i is the ranking of document x_i where$y_i < y_j$ if $x_i \succ x_j$. This means the rank of ith document is lower than the rank of jth document if ith document is preferred over jth document.

Output As mentioned above, output of a classification task is a class label out of a discrete set. While in ranking, the output is a ranking score where input documents can be ordered. The ranking function will output scores $F(x_i) > F(x_j)$ if $x_i \succ x_j$

3.2 Ranking SVM

Ranking SVM is one of the state of the art ranking models which surpasses other methods in performance. Assume there is a linear function F which can be used for ranking as follows:

$$\forall\{(x_i, x_j) : y_i < y_j \in D\} : F(x_i) > F(x_j) \Leftrightarrow w.x_i > w.x_j$$

w is the weight vector which will be learnt by the algorithm to make its input data concordant with the ordering in D. This is known to be a np-hard problem [2]. Herbrich et al. [3] introduced a mechanism to approximate this function using Support Vector Machine where a slack variable ξ_{ij} is used and the optimization problem aims to minimize the upper bound of the slack variable ξ_{ij} as follows:
minimize:

$$L(w, \xi_{ij}) = \frac{1}{2}w.w + C\sum \xi_{ij}$$

subject to:

$$\forall\{(x_i, x_j) : y_i < y_j \in D\} : w.x_i \geq w.x_j + 1 - \xi_{ij}$$

$$\forall(i, j) : \xi_{ij} \geq 0$$

Above constraints bound the solution of the optimization problem to satisfy ordering entries in training set D with minimal error. By minimizing w.w we can maximize the margin ($\frac{1}{||w||}$). C is the soft margin parameter. By rearranging the first constraint we can receive following:

$$w(x_i - x_j) \geq 1 - \xi_{ij}$$

This is the familiar constraint in structured SVM. Therefore classical SVM implementations can be extended to address ranking problems.

4 Experiments

Experiments were carried out using the Twitter Stance dataset provided in SemEval 2016 Stance Detection task A [9]. This is a supervised task where participants were given stance labelled data for 5 targets (Hillary Clinton, Atheism, Climate Change is a Real Concern, Feminist Movement and Legalization of Abortion). Table 1 contains details about dataset. Tweets are labelled as favor if it supports the given target, against if it disagrees with the target, neither if either agree or disagree cannot be inferred.

SVM^{rank} tool [5] was used for experiments. Stance labels was mapped to a numerical value (eg: favor =1, against =3, neither = 2) and fed to the model. We used variety of methods to represent Tweets using Natural Language Processing techniques. We reshuffled all test and training data together and created separate partitions for training (80 %), validation(10 %) and testing (10 %). Structure SVM with a linear kernel is used as the baseline model as well as the model to identify what features are best for Tweet representations. Table 2 contains results on test set for target Hillary Clinton when different feature sets were used with optimal hyper parameter C (found using evaluation set) on the baseline model.

Table 1: SemEval 2016 Task 6 - sub task A dataset

Target	Training				Test			
	total	favor%	against%	neither%	total	favor%	against%	neither%
Atheism	513	17.9	59.3	22.8	220	14.5	72.7	12.7
Climate Change is a Real Concern	395	53.7	3.8	42.5	169	72.8	6.5	20.7
Feminist Movement	664	31.6	49.4	19.0	285	20.4	64.2	15.4
Hillary Clinton	689	17.1	57.0	25.8	295	15.3	58.3	26.4
Legalization of Abortion	653	18.5	54.4	27.1	280	16.4	67.5	16.1

Table 2: Average F measures of baseline model on test set for different features

Feature	C	$F_{\text{avg}}(\frac{f_{\text{favor}}+f_{\text{against}}}{2})$	Kendall's τ
Wordunigram	10	**0.5832**	0.4266
char ngram(1,3)	0.0001	0.3356	nan
char ngram(1,4)	0.0001	0.3356	nan
char ngram(1,5)	0.0001	0.3378	0.1268
char ngram(2,5)	1	0.567	0.5656
char ngram(3,5)	0.1	0.5206	0.5401
word ngram(1,2)	1	0.4352	0.3890
word ngram(1,3)	1	0.3980	0.3430
word ngram(1,4)	10	0.3471	0.2840
char(2,5) word(1) ngram combined	0.1	0.567	**0.5656**
char(2,5) word(1,2) ngram combined	0.1	0.567	**0.5656**
char(2,5) word(1,3) ngram combined	0.1	0.567	**0.5656**
char(2,5) word(1,4) ngram combined	1	0.567	0.4500
char(3,5) word(1,2) ngram combined	0.1	0.4813	0.5281
GloVe embedding	0.000 01	0.3331	0.1307
Word2Vec embedding	0.000 01	0.3342	0.3103

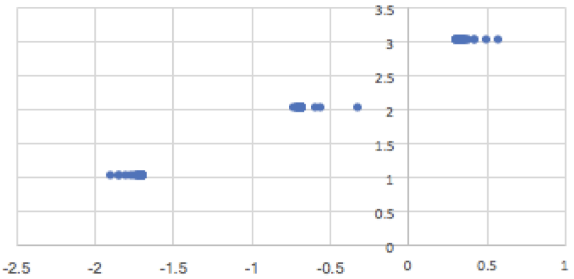


Figure 1: Ranking score distribution for training data (correlation = 0.75)

Average F measure of favor and against classes and correlation between predicted order of Tweets with ground truth were measured. Kendall's Tau was used for correlation measurement because it considers ordinal nature of data. Several observations were made from this. Character level from 2 to 5 was able to perform some significant impact on stance classification. In the word level, word unigrams showed the highest performance. And combined feature of word unigrams and bigrams came next. Therefore we created combined features using those best performing character and word level ngrams. Based on the char and word ngram combined features, green highlighted models performed well. Several types of label numbering such as (favor < against < neither and favor < neither < against) were evaluated to identify whether there is a significant difference among these ordering. There is a considerable amount of correlation difference based on the order. favor < against < neither gave superior results over favor < neither < against. But F measure did not change with the order. Above table shows correlation values.

Based on these observation it was found out that word and character ngram combined feature presentations give better results for this task. After having baseline model, we moved our experiments to Rank SVM. As mentioned before, output of this model is a list of scores. After ordering tweets based on the score and deciding class boundaries based on what data stay closed together in the ordering we can assign class labels to them. The expectation is that model should learn to assign similar score functions to same stance tweets as how they were given in the training set. We use correlation results of evaluation set to determine class boundaries. All those character and word ngram features gave similar results in Rank SVM. Therefore the correlation of feature set char(3,5) and word(1,2) gram is discussed here. Figure 1 shows how the scores of Ranking SVM for its training data is grouped. It is cleared that class boundaries can be easily drawn using it.

Figure2 shows how scores is distributed when validation data is sent to Rank SVM. This time there are know clear cut boundaries. Even not to a level which boundaries can be drawn in a way majority can be grouped together.

Several reasons can affect this poor performance of the model. Due to limited number of data, model might not have been able to capture actual ordering information. So it has over-fitted training data. These feature representations may not be suitable enough to capture correct information from the text.

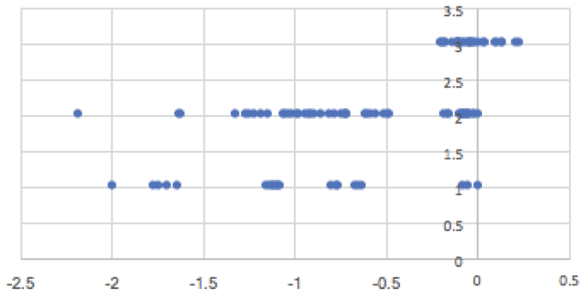


Figure 2: Ranking score distribution for validation data (correlation = 0.486)

5 Visual integration for Stance Detection

Due to the poor performance of above model and also considering the problems stance detection faced in the literature, we tried to think whether there is a way to get more information about author's stance from his post. This drew our attention towards images in posts.

Nowadays many users tend to use images and emoticons to express their ideas quite often. Therefore we identified the potential of using visual data to further enhance stance detection.

6 Conclusions

During this project period, we mainly focussed on analyzing existing work on stance detection and experiment on novel approaches. We tried to map stance detection as a ranking problem and address it accordingly. However we were unable to achieve superior performance to state of the art methods. This drew our attention to analyse and understand the potential of using visual content of user posts for stance detection as an additional information. This will further be analysed and experimented in the future.

7 Acknowledgement

We are very grateful for HPI Future SOC lab for offering us computational power for our experiments.

References

[1] H. Bøhler, P. Asla, E. Marsi, and R. Sætre. "IDI@NTNU at SemEval-2016 Task 6: Detecting Stance in Tweets Using Shallow Features and GloVe Vectors for

Word Representation". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 445–450. DOI: 10.18653/v1/S16-1072.

[2] W. W. Cohen, R. E. Schapire, and Y. Singer. "Learning to order things". In: *Advances in Neural Information Processing Systems*. 1998, pages 451–457. DOI: 10.1613/jair.587.

[3] R. Herbrich, T. Graepel, and K. Obermayer. *Large margin rank boundaries for ordinal regression*. 2000.

[4] Y. Igarashi, H. Komatsu, S. Kobayashi, N. Okazaki, and K. Inui. "Tohoku at SemEval-2016 Task 6: Feature-based Model versus Convolutional Neural Network for Stance Detection". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 401–407. DOI: 10.18653/v1/S16-1065.

[5] T. Joachims. "Training linear svms in linear time". In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2006, pages 217–226. DOI: 10.1145/1150402.1150429.

[6] P. Krejzl and J. Steinberger. "UWB at SemEval-2016 Task 6: Stance Detection". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 408–412. DOI: 10.18653/v1/S16-1066.

[7] C. Liu, W. Li, B. Demarest, Y. Chen, S. Couture, D. Dakota, N. Haduong, N. Kaufman, A. Lamont, M. Pancholi, K. Steimel, and S. Kübler. "IUCL at SemEval-2016 Task 6: An Ensemble Model for Stance Detection in Twitter". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 394–400. DOI: 10.18653/v1/S16-1064.

[8] A. Misra, B. Ecker, T. Handleman, N. Hahn, and M. Walker. "NLDS-UCSC at SemEval-2016 Task 6: A Semi-Supervised Approach to Detecting Stance in Tweets". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 420–427. DOI: 10.18653/v1/S16-1068.

[9] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry. "SemEval-2016 Task 6: Detecting Stance in Tweets". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 31–41. DOI: 10.18653/v1/S16-1003.

[10] B. G. Patra, D. Das, and S. Bandyopadhyay. "JU_NLP at SemEval-2016 Task 6: Detecting Stance in Tweets using Support Vector Machines". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 440–444. DOI: 10.18653/v1/S16-1071.

[11] A. Rajadesingan and H. Liu. “Identifying users with opposing opinions in twitter debates”. In: *International Conference on Social Computing, BehavioralCultural Modeling, and Prediction*. Springer, 2014, pages 153–160. doi: 10.1007/978-3-319-05579-4_19.

[12] P. Sobhani. “Stance Detection and Analysis in Social Media”. PhD thesis. Université d’Ottawa/University of Ottawa, 2017.

[13] S. Somasundaran and J. Wiebe. “Recognizing stances in online debates”. In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Volume Volume 1. Association for Computational Linguistics, 2009, pages 226–234. doi: 10.3115/1687878.1687912.

[14] M. Thomas, B. Pang, and L. Lee. “Get out the vote: Determining support or opposition from congressional floor-debate transcripts”. In: *Proceedings of the 2006 conference on empirical methods in natural language processing*. Association for Computational Linguistics, 2006, pages 327–335. doi: 10.3115/1610075.1610122.

[15] M. Tutek, I. Sekulic, P. Gombar, I. Paljak, F. Culinovic, F. Boltuzic, M. Karan, D. Alagić, and J. Šnajder. “TakeLab at SemEval-2016 Task 6: Stance Classification in Tweets Using a Genetic Algorithm Based Ensemble”. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 464–468. doi: 10.18653/v1/S16-1075.

[16] P. Vijayaraghavan, I. Sysoev, S. Vosoughi, and D. Roy. “DeepStance at SemEval-2016 Task 6: Detecting Stance in Tweets Using Character and Word-Level CNNs”. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 413–419. doi: 10.18653/v1/S16-1067. arXiv: 1606.05694 [cs.CL].

[17] M. A. Walker, P. Anand, R. Abbott, and R. Grant. “Stance classification using dialogic properties of persuasion”. In: *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2012, pages 592–596.

[18] W. Wei, X. Zhang, X. Liu, W. Chen, and T. Wang. “pkudblab at SemEval-2016 Task 6 : A Specific Convolutional Neural Network System for Effective Stance Detection”. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 384–388. doi: 10.18653/v1/S16-1062.

[19] M. Wojatzki and T. Zesch. “ltl.uni-due at SemEval-2016 Task 6: Stance Detection in Social Media Using Stacked Classifiers”. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 428–433. doi: 10.18653/v1/S16-1069.

[20] H. Yu and S. Kim. "Svm tutorialclassification, regression and ranking". In: *Handbook of Natural computing*. Springer, 2012, pages 479–506.

[21] G. Zarrella and A. Marsh. "MITRE at SemEval-2016 Task 6: Transfer Learning for Stance Detection". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 458–463. DOI: 10.18653/v1/S16-1074. arXiv: 1606.03784 [cs.AI].

[22] Z. Zhang and M. Lan. "ECNU at SemEval 2016 Task 6: Relevant or Not? Supportive or Not? A Two-step Learning System for Automatic Detecting Stance in Tweets". In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California: Association for Computational Linguistics, June 2016, pages 451–457. DOI: 10.18653/v1/S16-1073.

Towards a GPU-Accelerated Causal Inference

Christopher Schmidt and Johannes Huegle

Hasso-Plattner-Institute, Potsdam, Germany
{christopher.schmidt,johannes.huegle}@hpi.de

The emergence of the Internet of Things (IoT) allows for a comprehensive analysis of industrial manufacturing processes. While domain experts within the company have enough expertise to identify the most common relationships, they will require support in the context of both, an increasing amount of observational data and the complexity of large systems of observed features. This gap can be closed by machine learning algorithms of causal inference that derive the underlying causal relationships between the observed features. Based on the method's high computational complexity we investigate the application of Graphics Processing Units (GPUs) to develop an efficient implementation for Gaussian distributed data. In our work, we evaluate a GPU-accelerated implementation for the calculation of the correlation matrix utilizing shared memory to achieve a speedup of up to 1.5 compared to an existing CUDA-enabled version and up to 2 compared to an efficient version executed on a CPU.

1 Introduction

Throughout the last decades, the rise of IoT led to a growing interest into the analysis of massive and complex datasets collected in the context of industrial manufacturing processes. The knowledge about the relational structure of the observed features allows to derive actionable insights, e.g., for the detection, prediction, and avoidance of machine failures, or the identification of root causes for discarded manufactured goods.

While the usage of sensors enables to collect data that monitors the whole manufacturing process it still remains to domain experts to determine the basic relational structures. With an increasing amount of influencing features, and a rising complexity in the industrial manufacturing world the identification of important relationships requires algorithmic support [12]. Algorithms for causal inference, e.g., see [19], use conditional independence (CI) tests to receive information about underlying relationships. Building on this skeleton, the algorithms determine the orientation of the detected relationships to construct a causal graphical model. The resulting graph involves nodes that depict the observed features and directed edges, which represent a causal relationship of the corresponding features within the manufacturing process.

The selection of the appropriate CI tests is directly determined by the underlying distribution of the considered features [2]. In the context of IoT, the Gaussian distribution with linear relationships is an often considered sensor distribution model

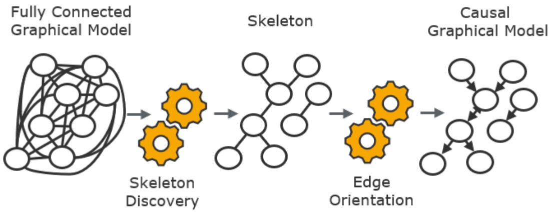


Figure 1: A schematic representation of the causal inference procedure

[11]. Under this sensor distribution model, the corresponding CI tests are based on the partial correlations of the involved features. In the worst case, the runtime of the algorithm is exponential to the number of nodes such that the inefficiency of the algorithm hinders its application in practise [7].

To address this drawback, we investigate the applicability of GPUs for the causal inference procedure. GPUs have been proven to be suitable execution devices for computational expensive machine learning algorithms, e.g., deep learning [4] or enterprise simulations [17]. Devices, such as the NVIDIA K80 GPU, reach a peak performance of up to 8.74 TFLOPS, are equipped with 24 GB of on-chip high bandwidth memory and outperform Central Processing Units (CPUs) [13]. Achieving peak performance requires a data parallel task, which matches the SIMT [10] execution model of a GPU. We aim to harness the processing power of the GPU to address the algorithm's computational complexity. In particular, we address the calculation of the correlation matrix for an underlying Gaussian distribution model.

The remainder of this report is organized as follows. Section 2 provides an introduction into the concept of causal inference in the context of the Gaussian distribution model. In Section 3, we present our current CUDA-based implementation for the calculation of the correlation matrix including preliminary results. Furthermore, we discuss first ideas of a CUDA-based parallel skeleton graph implementation in Section 4, and summarize our work in Section 5.

2 Causal Inference Procedure

Due to the work of Judea Pearl [15] and Spirtes et al. [18] the notion of causality has grown from a nebulous concept into a mathematical theory based on the probabilistic graphical modeling.

In this framework, the causal relationships are represented by the causal graphical model $\mathcal{G} = (F, E)$ that involves a finite set of N nodes $F = (F_i)_{i \in \mathcal{F}}$, $\mathcal{F} = \{1, \dots, N\}$, each representing the observed feature $F_i \in F$, and set of directed edges E. Here, a direct edge $F_i \to F_j$ depicts a direct causal relationship from F_i to F_j, $i, j \in \mathcal{F}$, $i \neq j$, that exists if changing the value of F_i results in changes in the distribution of F_j, assuming that the values of all other variables in F $\{F_i, F_j\}$ are fixed.

A conceptual algorithm for learning the causal graphical model was introduced by Spirtes et al. [19]. As depicted in the schematic representation in Figure 1, the so called PC algorithm operates in two phases.

The first phase, the skeleton discovery, begins with a fully connected undirected graph of the observed features F, and determines whether an edge $F_i — F_j$ of adjacent features F_i and F_j, $i, j \in \mathcal{F}$, $i \neq j$, should be removed. Therefore, the algorithm checks for the conditional independence of all pairs of adjacent features F_i and F_j, given an increasing separation set $S \subseteq F\ \{F_i, F_j\}$, $i, j \in \mathcal{F}$, denoted by

$$F_i \perp\!\!\!\perp F_j \mid S.$$

In particular, for each level regarding the current cardinality of the separation set S the undirected edge
$F_i — F_j$ is kept if and only if the null hypothesis F_i *and* F_j *are conditionally independent* is rejected at significance level α for all separation sets $S \subseteq F\ \{F_i, F_j\}$, $i, j \in \mathcal{F}$. Once the pair of features F_i, F_j, $i, j \in \mathcal{F}$, is found to be conditional independent for the current level the edge between F_i and F_j is removed, the corresponding separation set S is persisted, and the cardinality of S is increased for the next level of CI tests.

The second phase of the algorithm starts with the skeleton produced in the first phase and aims to orient as many undirected edges as possible. Therefore, for non-adjacent features F_i and F_j with common neighbour F_k, and unequal $i, j, k \in \mathcal{F}$ apply the following rule $R1$ to orient all the colliding edges in the skeleton:

R1. Replace $F_i - F_j - F_k$ by $F_i \rightarrow F_j \leftarrow F_k$ if and only if F_j is not in the corresponding separation set S.

In the resulting partially oriented graph, for all unequal $i, j, k, l \in \mathcal{F}$, the following four rules $R2$-$R5$ are applied repetitively until no more undirected edges can be oriented:

R2. Orient $F_j — F_k$ into $F_j \rightarrow F_k$ whenever there is a directed edge $F_i \rightarrow F_j$ such that F_i and F_k are nonadjacent.

R3. Orient $F_i — F_j$ into $F_i \rightarrow F_j$ whenever there is a chain $F_i \rightarrow F_k \rightarrow F_j$.

R4. Orient $F_i — F_j$ into $F_i \rightarrow F_j$ whenever there are two chains $F_i — F_k \rightarrow F_j$ and $F_i — F_l \rightarrow F_j$ such that F_k and F_l are nonadjacent.

R5. Orient $F_i — F_j$ into $F_i \rightarrow F_j$ whenever there are two chains $F_i — F_k \rightarrow F_j$ and $F_k \rightarrow F_l \rightarrow F_j$ such that F_k and F_l are nonadjacent.

Note, that Kalisch et al. [3] had proven the uniform consistency of the PC algorithm in our sensor distribution model with a very high-dimensional, sparse Gaussian distribution. This implies that the algorithm consistently estimates the underlying causal graphical model as the sample size increases.

Moreover, in our sensor distribution model of Gaussian distributed data $F_i \perp\!\!\!\perp F_j \mid S$ reduces to zero partial correlation between F_i and F_j given the separation set S, with $S \subseteq F\ \{F_i, F_j\}$, $i, j \in \mathcal{F}$, $i \neq j$, e.g., see [6]. Thus, following the ideas of Kalisch and Bühlmann [3] we can implement a computationally feasible CI testing procedure based on testing whether the corresponding partial correlation is zero or not. Furthermore, the partial correlations can be computed efficiently by inverting the correlation matrix Cor [16].

3 Correlation Matrix

As described in Section 2, a computationally feasible causal inference procedure can be build upon the correlation matrix Cor. Nevertheless, the calculation of the correlation matrix bundles a big part of the algorithm's computational complexity such that it is worth to examine hardware acceleration techniques to speed up the calculation.

Recall, that the Pearson correlation coefficient $\rho_{i,j}$ for a given pair of observed features F_i and F_j, $i, j \in \mathcal{F}$, with n observations $n < \infty$ is defined as

$$\rho_{i,j} = \frac{\sum_{s=1}^{n} \left(F_i^{(s)} - \overline{F_i}\right)\left(F_j^{(s)} - \overline{F_j}\right)}{\sqrt{\left(F_i^{(s)} - \overline{F_i}\right)^2}\sqrt{\left(F_j^{(s)} - \overline{F_j}\right)^2}} \tag{1}$$

with the arithmetic means $\overline{F_i}$, and $\overline{F_j}$ of the corresponding features F_i, and F_i, i.e.,

$$\overline{F_i} = \sum_{s=1}^{n} F_i^{(s)}, \ \overline{F_j} = \sum_{s=1}^{n} F_j^{(s)},$$

where $F_i^{(s)}$, and $F_j^{(s)}$ denotes the s-th entry of the feature F_i, and F_j, respectively, e.g., see [9]. As $\rho_{i,i} = 1$, for all $i \in \mathcal{F}$, the Pearson correlation coefficient $\rho_{i,j}$ has to be calculated for all pairs of features F_i and F_j, with $i, j \in \mathcal{F}$, $i \neq j$. This generates the N-dimensional symmetric correlation matrix Cor, where each entry $\text{Cor}_{i,j}$ of row i and column j represents the Pearson correlation coefficient $\rho_{i,j}$ of the corresponding features F_i and F_j, $i, j \in \mathcal{F}$. Note, by averaging the sums in Equation 1 the correlation coefficient $\rho_{i,j}$ can also be derived using the standard deviations of the corresponding features F_i, F_j, $i, j \in \mathcal{F}$, respectively.

The `gputools` R package [1] provides a CUDA-enabled implementation to calculate the correlation coefficients. The authors of the library make use of shared memory and approach parallelism for calculating mean $\overline{F_i}$ and standard deviation σ_i by creating a single thread block per feature F_i, $i \in \mathcal{F}$, leading to N thread blocks in total. The threads within a thread block reduce the observational data to calculate the corresponding values. During the calculation of the correlation coefficient a separate thread block is started per entry $\text{Cor}_{i,j}$, $i, j \in \mathcal{F}$, of the correlation matrix, which results in the execution of N^2 thread blocks. Based on this implementation the observational data of one feature is read $2 * N$ times from global memory.

In our approach, we aim to reduce the number of accesses to global memory. Instead of starting a new thread block per calculation of the correlation coefficient we adapt the implementation to calculate multiple correlation coefficients in one thread block. Ideally, for feature F_i,$i \in \mathcal{F}$, we want to calculate the entries $\text{Cor}_{i,j}$ for all $j \in$

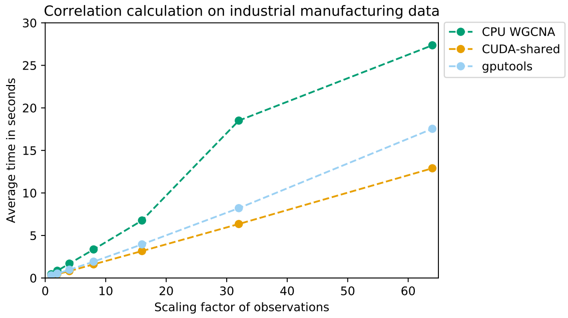


Figure 2: Execution time of the correlation calculation for an increasing number of observations

$\mathcal{F} \{i\}$ in one thread block. This enables to read the observational data of one feature F_i,$i \in \mathcal{F}$, only once and reduces the number of reads from global memory to N. As we need to store partial results of the threads within a thread block in shared memory this approach is limited by the amount of available shared memory. The Kepler GK210 GPU Computing architecture allows to use up to 48 KB of shared memory per thread block and allows to have up to 16 thread blocks per Streaming Multiprocessor (SM) [14]. Based on these characteristics and an experimental evaluation varying the number of threads per thread block we determined that calculating 4 entries within each thread block proved to provide the most speed-up and is used for our evaluation.

In a first experiment, we compare our implementation of the calculation of the correlation coefficient, which we call `CUDA-shared` to the CUDA-enabled version from the R package `gputools` [1] and to an efficient CPU based version from the R package `WGCNA` [5]. Our results are based on a small dataset abstracting an industrial manufacturing use case with 354 features and over 38, 000 observations. We enlarge the dataset size by a scaling factor to investigate how the implementation works with an increasing amount of observations, as this is more realistic for a sensor-based use case. The results of the first experiment are presented in Figure 2, and show that the execution time of the calculation of the correlation matrix scales linearly with the number of observational data values. Our implementation, `CUDA-shared`, provides a speed-up compared to the other two implementations and outperforms the `gputools` version of up to factor 1.3 for an industrial manufacturing dataset.

In a second experiment, we compare the two CUDA-enabled implementations on different real world datasets, which have been used by the authors of the `ParallelPC` algorithm [8] to evaluate their algorithm for causal inference. In Table 1, we can see that our implementation `CUDA-shared` speeds up the execution of the calculation of correlation matrix for a range of real world datasets. In particular, we achieve a speed up of factor 1.58 for the dataset `Scerevisiae` that incorporates 2, 810 features and 160 observations.

For the experiments we used an NVIDIA K80 GPU provided by the Future SOC Lab and an Intel i7-6700K CPU with 4 cores.

Table 1: Calculation of the correlation matrix gputools vs CUDA-shared - median of 100 executions in milliseconds

	gputools	CUDA-shared
BR51	44.79	29.07
MCC	35.17	23.83
NCI-60	25.18	16.98
Saureus	184.13	124.89
Scerevisiae	564.83	357.46

4 Skeleton Discovery

The calculation of the correlation matrix, for which we provide an improved implementation, as shown in Section 3, is our first step to towards a GPU-accelerated causal inference algorithm. As described in Section 2, the resulting correlation matrix serves as the basis of the skeleton discovery, for which we aim to develop an improved implementation in future work. Based on the work from Le et al. [8], and our experience from the implementation of `CUDA-shared`, we consider two different strategies for a CUDA-enabled skeleton discovery. The main goals are to achieve a high occupancy, fully utilizing the parallel computational power of a GPU and to reduce unnecessary accesses to global memory. To address these objectives we propose to process individual CI tests for each combination of adjacent features F_i, F_j, $i, j \in \mathcal{F}$, $i \neq j$, within a single level regarding the cardinality of the separation set $S \subseteq \mathcal{F}\ \{F_i, F_j\}$ in parallel. This strategy should work for a dataset with a large number of features, within the first two levels, as the number of CI tests to be conducted is high enough to fully occupy the processing units of the GPU. Depending on the number of identified relationships within a level, the number of CI tests to be performed in the subsequent level will gradually decrease, while the computational effort for each individual Gaussian CI test increases. Thus, we also see the potential to provide a CUDA-enabled version for the calculation of a single CI test. In our future work, we plan to develop both versions, investigate their suitability for the GPU and compare their performance on real world datasets.

5 Summary

In this project report, we shared our current status of our work towards a GPU-accelerated causal inference. We provided an overview on the concepts of causal inference in the context of the Gaussian distribution model, and proposed an improved CUDA-enabled calculation of the correlation matrix. Our implementation reduces the number of accesses to global memory, by utilizing the available shared memory. The empirical results showed that our implementation speeds up an existing CUDA-enabled algorithm and outperforms an efficient CPU-based version on real world datasets. Based on these results, we are confident that we can address other parts of the causal inference procedure and provide GPU-accelerated implementations.

References

[1] J. Buckner, M. Seligman, F. Meng, and J. Wilson. *gputools: A Few GPU Enabled Functions*. 2016.

[2] A. P. Dawid. "Conditional independence in statistical theory". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1979), pages 1–31.

[3] M. Kalisch and P. Bühlmann. "Estimating high-dimensional directed acyclic graphs with the PC-algorithm". In: *Journal of Machine Learning Research* 8.Mar (2007), pages 613–636.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems 25*. Edited by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., 2012, pages 1097–1105.

[5] P. Langfelder and S. Horvath. "WGCNA: an R package for weighted correlation network analysis". In: *BMC Bioinformatics* 9.1 (Dec. 2008), page 559. ISSN: 1471-2105. DOI: 10.1186/1471-2105-9-559.

[6] A. Lawrance. "On conditional and partial correlation". In: *The American Statistician* 30.3 (1976), pages 146–149.

[7] T. Le, T. Hoang, J. Li, L. Liu, H. Liu, and S. Hu. "A fast PC algorithm for high dimensional causal discovery with multi-core PCs". In: *IEEE/ACM transactions on computational biology and bioinformatics* (2016).

[8] T. D. Le, T. Hoang, J. Li, L. Liu, and H. Liu. "A fast PC algorithm for high dimensional causal discovery with multi-core PCs". In: *CoRR* abs/1502.02454 (2015).

[9] J. Lee Rodgers and W. A. Nicewander. "Thirteen ways to look at the correlation coefficient". In: *The American Statistician* 42.1 (1988), pages 59–66.

[10] E. Lindholm, J. Nickolls, S. Oberman, and J. Montrym. "NVIDIA Tesla: A Unified Graphics and Computing Architecture". In: *IEEE Micro* 28.2 (Mar. 2008), pages 39–55. ISSN: 0272-1732. DOI: 10.1109/MM.2008.31.

[11] R. C. Luo, M.-H. Lin, and R. S. Scherp. "Dynamic multi-sensor data fusion system for intelligent robots". In: *IEEE Journal on Robotics and Automation* 4.4 (1988), pages 386–396.

[12] K. Marazopoulou, R. Ghosh, P. Lade, and D. Jensen. "Causal Discovery for Manufacturing Domains". In: *arXiv preprint arXiv:1605.04056* (2016).

[13] NVIDIA Corporation. *NVIDIA Tesla K80 The World's fastest GPU Accelerator*. Nov. 2014.

[14] NVIDIA Corporation. *NVIDIA's Next Generation CUDA Compute Architecture: Kepler GK110/210*. 2014.

[15] J. Pearl. *Causality: Models, Reasoning and Inference*. 2nd. New York, NY, USA: Cambridge University Press, 2009. ISBN: 978-0-521-89560-6.

[16] J.-P. Pellet and A. Elisseeff. "A Partial Correlation-Based Algorithm for Causal Structure Discovery with Continuous Variables". In: (Sept. 2007), pages 229–239.

[17] C. Schwarz, C. Schmidt, M. Hopstock, W. Sinzig, and H. Plattner. "Efficient Calculation and Simulation of Product Cost Leveraging In-Memory Technology and Coprocessors". In: *The Sixth International Conference on Business Intelligence and Technology (BUSTECH 2016)*. 2016.

[18] P. Spirtes. "Introduction to causal inference". In: *Journal of Machine Learning Research* 11.May (2010), pages 1643–1662.

[19] P. Spirtes, C. N. Glymour, and R. Scheines. *Causation, prediction, and search*. MIT press, 2000.

Analytical environment for high-speed user behaviour outlier detection

Andrey Sapegin, David Jaeger, Feng Cheng, and Christoph Meinel

Hasso Plattner Institute, Potsdam, Germany
{firstname.lastname}@hpi.uni-potsdam.de

Modern SIEM systems include various tools and algorithms for analysis of security-related events in the enterprise network. These systems utilise hybrid detection techniques, which implement signature- and query-based detection, as well as anomaly detection methods. However, the efficient analysis of user behaviour often requires application of specially designed methods, due to the fact that some malicious user activities do not result in authentication failures and, therefore, do not trigger access control signatures. In this report we describe a SIEM system module for deep user behaviour analytics, which is capable to efficiently detect various deviations from modelled user behaviour in the enterprise network.

1 Introduction

The analysis of user behaviour remains to be a complicated task for security experts. To detect different types of malicious user behaviour, such as intrusion without access violation, special methods need to be implemented [7]. These methods are usually based on the anomaly detection approach. To perform the anomaly detection, the model of the normal user behaviour is built to identify deviation from it. Other implementations of anomaly/outlier detection are based on the clustering of the user data and select suspicious user events that are located far from the cluster center. In this report, we present an example of the detailed analysis of user data from large multinational company. The selected dataset includes the activity of privileged users during a period of one months. To perform the efficient detection of suspicious user behaviour, the user activity data are enriched using other sources that contain information about user groups, location of the source and destination for user-initiated connections, as well as the user office locations. Next, we analyse the enriched data with various high-speed outlier detection methods.

2 HPI Future SOC Lab resources

The analysis of user behaviour was performed on the hardware provided by HPI Future SOC Lab (in shared access mode). The data were stored and preprocessed in the two SAP HANA [4] database instances, whereas the outlier detection was performed using Dell DL-980 server with 256 GB RAM and 64 CPU cores, where the VMware ESXi hypervisor was installed.

The environment for security log analytics as well as the algorithms utilised for analysis of user behaviour (outlier detection) are described in details in the next section.

3 Analytical Environment

The analysis of user behaviour is performed in the proof-of-concept SIEM system being developed at Hasso Plattner Institute: Real-time Event Analytics and Monitoring System (REAMS), which was described in the previous report (for the first phase of this project) [8]. The architecture of REAMS is shown unchanged in the Figure 1.

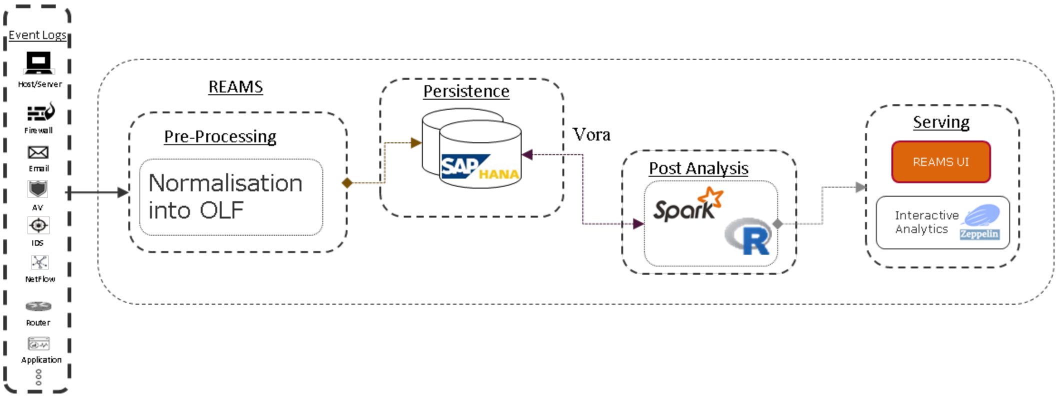


Figure 1: Architecture of Real-time Event Analytics and Monitoring System

To perform the analysis of user behaviour, a number of algorithms was executed in R (through SAP HANA R Integration [5]) and Spark [1] (through SAP Vora [6]), as described in the section below.

4 Outlier detection for user behaviour analytics

During the current phase of the HPI Future SOC Lab project, we have utilised two high-speed algorithms developed earlier (User Behaviour Outlier Detection [7] and Hybrid Outlier Detection [9]).

In addition to existing algorithms, we have developed the outlier detection based on the the Map Reduce implementation of k-means algorithm in Apache Spark. To reach high performance, all steps of the algorithm, such as feature preprocessing, selection and scaling (and not only clustering itself) are implemented using Map Reduces and are highly scalable.

The features for the Map Reduce outlier detection are listed below:

- number of logon events
- number of failed logon events
- number of file share accesses
- number of unique failure reasons
- number of destination countries
- number of departments the destination belongs to
- number of source countries
- number of departments the source belongs to
- number of computers on which the user performed an interactive logon

Using these features, the proposed outlier detection algorithm is able to detect both users with most unusual values for single features and users with unusual combinations of feature values (e.g., users with low number of failed logon events, but different failure reasons).

All mentioned algorithms were applied on the dataset, which is described in the next section.

5 Windows Events dataset

The Windows Events dataset was provided from our enterprise partner and includes activity of the privileged users during January 2017. The dataset includes only Windows Events with Event ID related to the network activity of users, namely Event IDs 4624 (successful logon), 4625 (failed logon) and 5140 (share access). The distribution of these event types is shown in Figure 2.

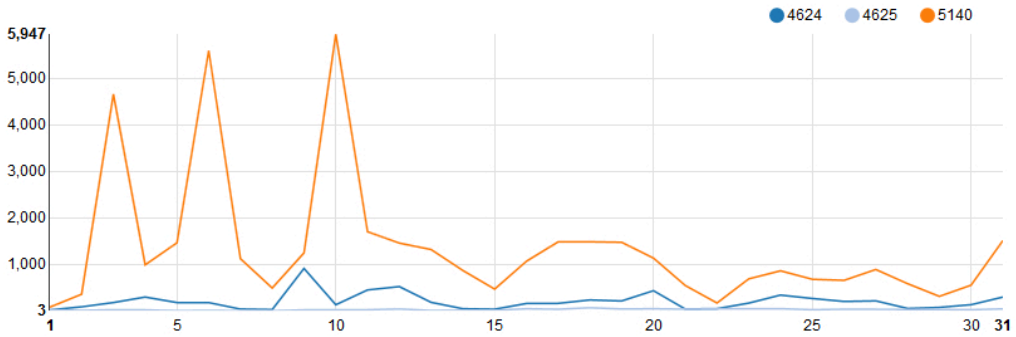


Figure 2: Network activity of privileged users in Windows Event data

Figure 2 shows the daily activity of privileged users during January 2017. The file share activity was rather high in the first three weeks (due to an internal schedule of the company), but then stabilised starting from the middle of the month. To provide a better view on the user activity, the number of active users per day is shown in Figure 3

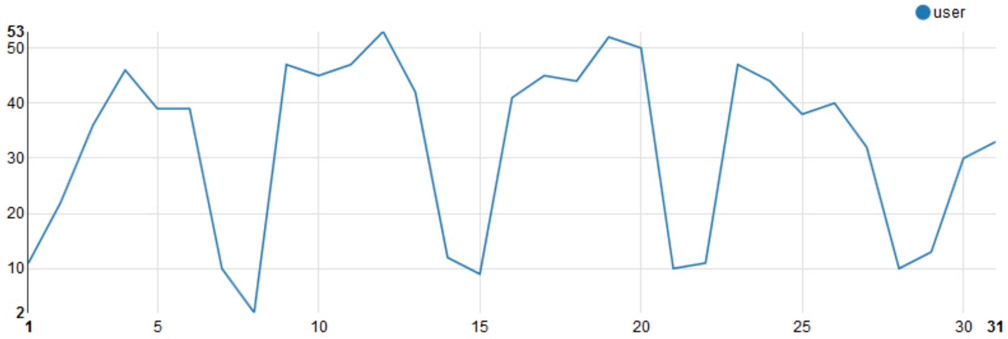


Figure 3: Number active privileged users per day

In Figure 3 the number of active users follows the week schedule with drops on the weekend.

To model this user activity and analyse logon events, as well as connections to the file shares, we correlate Windows Events with the data feed containing information about enterprise assets (including user's department, office location, locations of company's servers, etc) as described in the subsection below. To perform this correlation, the dataframes with Windows Events and company's assets are joined using pyspark in the Zeppelin [2] environment.

The results of outlier detection on the correlated dataset are presented in the next section.

6 Outlier detection results

Using the User Behaviour Outlier Detection [7], we detected two types of outliers:

- Events with low probability, which implies that the particular type of user activity (connection to specific server or file share) was modelled as normal, but it became outlier due to some properties, e.g. unusually high number of authentication failures within the time interval. This type of outliers included 2 users that triggered failed logon events at the strange time (in the night) on multiple servers.
- Events with zero probability that cover the cases, where the type of the user activity was not modelled as normal during the training phase. Looking on

the list with such outliers, we discovered 3 users that connected to unexpected destinations all over the world (6–8 destination countries per user, which is not an expected user behaviour pattern at the partner enterprise).

Next, the Hybrid Outlier Detection [9] also highlighted users with high number of authentication failures and connections to server in lange number of countires. Besides this, two detected outliers were unique to this type of outlier detection:

- A number of users from the same department that have up to 10 interactive logon events (using keyboard) per second on the same Windows Domain Controller. Such user behaviour could be result of custom scripts that will be investigated by our enterpriese partner.
- A number of users with very similar usernames (including the same first and last name of single person) with very high percentage of failed logon events (50 %) with different failure reasons ("The user has not been granted the requested logon type", "Unknown user name or bad password", "account locked out", "The specified account has expired", etc.) and on multiple servers.

Finally, using newly developed Map Reduce k-means-based outlier detection, we were able to detect 2 users that were similar to the second type of outliers from User Behaviour Outlier Detection. These users connected to the servers all over the world (16–38 different countries) and had high number of ailed logon events (up to 34 %of total amount of events for the particular user).

7 Conclusion

This report covered the second phase of the HPI Future SOC Lab project (for details about the first phase please see the previous report [8]). Thanks to the hardware and hardware resources provided by the Future SOC Lab we were able to setup the Analytical Environment and perform outlier detection on two different Windows Event datasets (mainly covering user behaviour). The proposed high-speed outlier detection methods comprehend each other and allow to detect various types of outliers, which proves the relevance of the developed approaches for the modern SIEM systems.

8 Future work

For the future work, we plan to further extend our analytical environment and implement new methods for high-speed analysis of security-related events. Besides advanced outlier detection approaches, we plan to utilise graph-based analytical methods, similar to the one proposed in [3]. The graph-based analytics allows to incorporate different types of data feeds in one graph and analyse interconnections between different entities, such as domain names, servers, users, etc. Taking Threat

Intelligence data as ground truth, such graphs can be utilised to calculate the probability of being related to malicious activity for each entity, which can be used to further improve the detection capabilities of our analytical environment.

References

[1] *Apache Spark - Lightning-fast luster computing*. URL: https://spark.apache.org/ (last accessed 2017-01-01).

[2] *Apache Zeppelin. A web-based notebook that enables interactive data analytics.* URL: https://zeppelin.apache.org/ (last accessed 2017-01-01).

[3] P. Najafi, A. Sapegin, F. Cheng, and C. Meinel. "Guilt-by-Association: Detecting Malicious Entities via Graph Mining". In: *13th EAI International Conference on Security and Privacy in Communication Networks (SecureComm 2017)*. 2017.

[4] *SAP HANA*. URL: http://www.saphana.com (last accessed 2017-01-01).

[5] *SAP HANA R Integration Guide*. URL: https://help.sap.com/hana/SAP_HANA_R_Integration_Guide_en.pdf (last accessed 2017-01-01).

[6] *SAP Vora*. URL: https://www.sap.com/product/data-mgmt/hana-vora-hadoop.html (last accessed 2017-01-01).

[7] A. Sapegin, A. Amirkhanyan, M. Gawron, F. Cheng, and C. Meinel. "Poisson-based anomaly detection for identifying malicious user behaviour". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Volume 9395. 2015, pages 134–150. ISBN: 978-3-319-25743-3. DOI: 10.1007/978-3-319-25744-0_12.

[8] A. Sapegin, D. Jaeger, F. Cheng, and C. Meinel. *Security analytics of large-scale heterogeneous data*. HPI Future SOC Lab Report. Apr. 2017.

[9] A. Sapegin, D. Jaeger, F. Cheng, and C. Meinel. "Towards a system for complex analysis of security events in large-scale networks". In: *Computers & Security* 67 (2017), pages 16–34.

Modelling of aluminum reduction processes with machine learning approaches and mathematical models using SAP HANA in-memory computing for smart-devices

Roberto C. L. de Oliveira and Fábio M. Soares

Federal University of Pará, Brazil
{limao,fms}@ufpa.br

Metallurgical modelling based on the physical laws (white box) purely covers all the theoretical phenomena in the plant, but its development takes too much effort, and moreover the model should be tuned against real data, which are often noisy. On the other hand, a purely data driven modelling (black box) requires less effort and provides a simpler model, but includes a significant bias as the plant history is the only foundation of this kind of model. Since a good process model should be prepared to reproduce any behavior a plant might present, even if it never happened, we thought of a hybrid modelling (grey box) considering the response of theoretical models along with the actual data to find simple process models that can be realized in small smart-devices. However, a lot of processing and exhaustive optimization is required in these tasks, making mandatory the use of parallel computing and in-memory facilities. The result of this work is to develop an approach to model a complex problem like Aluminum Reduction using the physical already known models as data generators to add further data into the existing dataset, which should be used a source for machine learning algorithms.

1 Introduction and background

The metallurgy industry involves several disciplines and competencies. In addition, production cells are often subject to disturbances and unexpected deviations, making the process control an even harder challenge [4]. Process models help in forecasting variables, giving trends, simulation of what-if scenarios and tests as well [1, 5, 7]. An accurate model depends on decent quality data covering a range of process operations as wide as possible. However, data-driven modeling (black box) often hinders this range, because plants usually work under closed loop, therefore representing only the plant's steady state [2, 3, 6]. On the other hand, a physical modelling (white box) is too theoretical, sometimes involving nearly unsolvable equations or calculations that would take a lot of effort, and often the models need a fine-tune to adjust to real data [6]. Keeping both paradigms in mind, it is encouraged the use of both approaches in building a stronger model combining their good characteristics together. The white box models can be used to produce complementary data to help in the building of a black box model. This fact encourages use of high performance com-

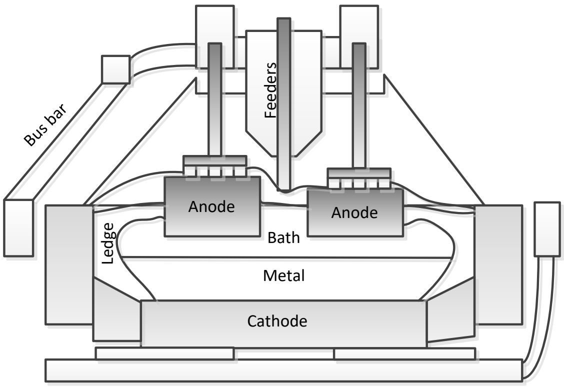


Figure 1: A typical aluminum reduction cell

puting resources, like parallelism and in-memory computing. What motivated this work was the possibility of using the FSOC powerful hardware to perform massive calculations jointly with the SAP HANA® in-memory capability.

2 A brief description of the process

Aluminum is produced by the Hall-Héroult process, an electrolysis based production by which aluminum is extracted from the alumina molecules under a strong electric current [1, 4, 5, 7]. This reaction occurs in the electrolyte also known as bath. The bath is a mixture of several chemical species which in turn causes variations in the bath temperature as well as in the physical state of the bath. Figure 1 shows a schema of a typical production cell.

In the top of the cell there is a large bar conducting the electric current of about 150 A to 400 A. This current is directed downwards passing a set of anodes (positive pole), then crossing the bath and the metal produced gets down to the bottom as the current flows to the cathode (negative pole). The electric current causes a great heating of the bath making the reactions possible. However, a good production typically occurs in ranges under 975 °C because otherwise occurs the back reaction, i.e. aluminum reacts again with alumina and carbon, thereby turning into alumina again [1, 4, 5, 7].

- Production reaction: $2Al_2O_3 + 3C^+ \rightarrow 4Al^+ + 3CO_2$
- Back reaction: $2Al^+ + 3CO_2 \rightarrow Al_2O_3 + 3CO$

The stability in the factors is essential for a maximum efficiency. Important variables, according to the production are Cell Resistance, Cell Temperature, Metal purity and Chemical Concentration of species, among others. The ledge, also known as solid bath, plays a key role in the processes insofar it protects the cell against the bath itself and helps in the thermal and mass balance, so not only the chemical in the bath is

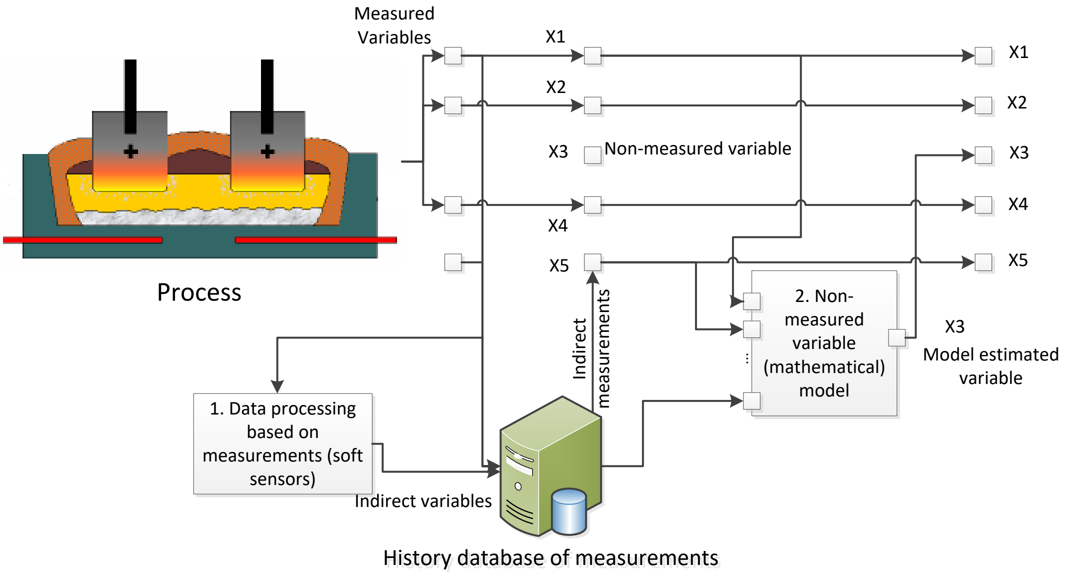

Measured Variables
X1
X2
X3
Non-measured variable
X4
X5
X1
X2
X3
X4
X5
Process
Indirect measurements
2. Non-measured variable (mathematical) model
X3
Model estimated variable
1. Data processing based on measurements (soft sensors)
Indirect variables
History database of measurements

Figure 2: Schema on how mathematical models are used to complete the process database

important but also in the ledge. For each of these variables there is a set point range, however under certain conditions this setpoint may be flexible, according to the cell status.

Aluminum smelters usually have a large number of cells, which in their turn are divided into sections, rooms and lines (also known as potlines or reductions) [4]. Provided that many phenomena are not yet fully understood by smelters, not every cell present the same behavior. In this sense every cell needs a specific control considering their peculiarities as well as their own parameters. Control systems collect a lot of data from a lot of cells, making the process database very huge, nevertheless these measurements are still just a small part of what happens in the process, since many variables (anode-to-cathode distance, alumina consumption and ledge composition) cannot be measured. On the other hand, with so many modeling works available in the literature, these variables can be estimated. In addition, not every variable can be measured every time, and some variables are measured only on-demand.

3 Method applied

The concerned approach consists of two steps. First is to expand the original database to include other process variables and the estimates when they were not measured.

In figure 2, five variables (X1 to X5) are shown, from which only X1, X2 and X5 are directly measured. A model (block #1) can be used to provide an indirect measure of X5 whenever there are not measures of this variable. For X3 a mathematical model (block #2) is applied to provide its measures, considering also measures of X5 which

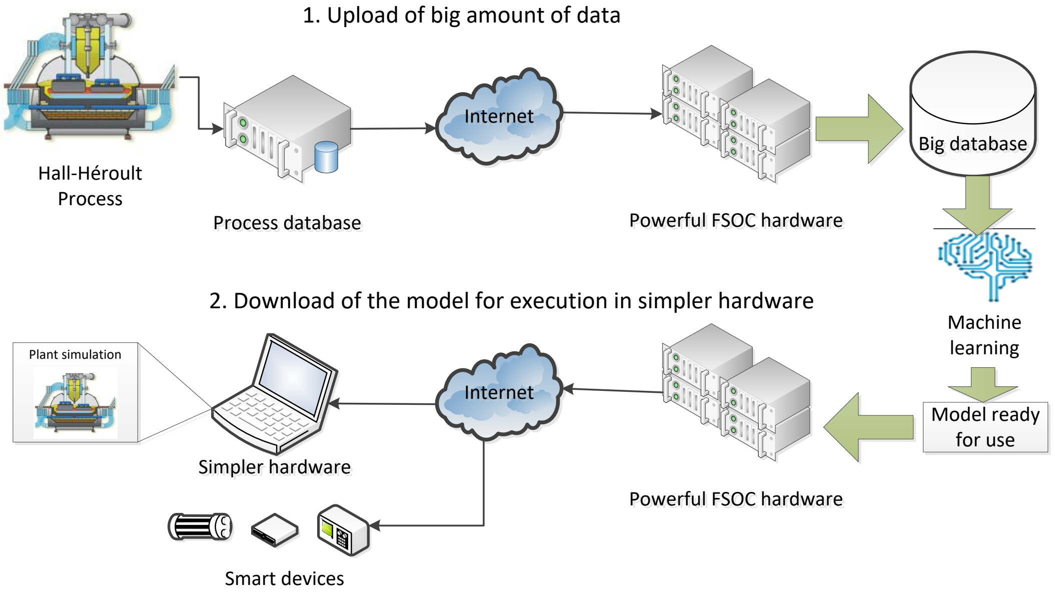


Figure 3: Overall process for building the models, from uploading the data (upper line # 1), building the model using machine learning approaches, to downloading the final model for use in laptops and smart devices.

were not present in the original database. At the end, all the data from the concerned variables will be present in the database.

The step 2 is to use machine learning techniques on the full process database. In this phase the model to be built should cover a wider operating range, that means the data should be selected considering all the operating range, however outliers should be dropped. Subsequently, using in-memory and parallel processing of SAP HANA Analytics plugin we exhaustively train several supervised machine learning algorithms that are suitable for executing in cheaper hardware such as IIoT smart devices.

4 Development of the environment

For this project we use eight virtual machines of the Future SOC Lab to run the mathematical models to generate all the data needed. The data are saved in a SAP HANA instance where the machine learning models are built upon the data using the in-memory processing. The virtual machines are divided into two groups: large (L) and extra large (XL). The large group processes potline variables (those which are common for all cells in a potline) and the extra large group processes cell variables (those which are specific for each cell). Each group has four virtual machines, one for each potline data we collected from a smelter. The mathematical models are implemented in Python 3.5 language and run in each of the virtual machines.

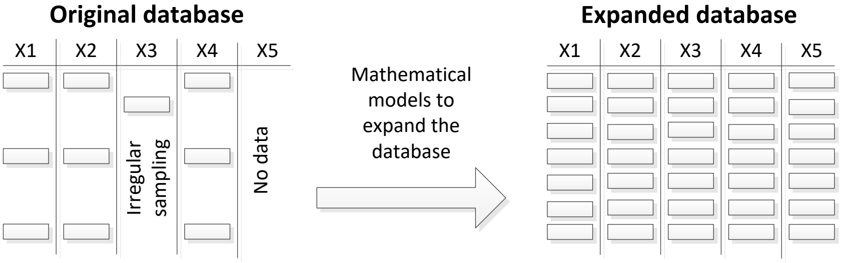


Figure 4: Database expansion

While modeling in the SAP HANA AFL environment we run several supervised learning algorithms (e.g. Linear Model, Support Vector Machines, Neural Networks), to find simpler models which can reproduce the behavior of the cells satisfactorily accurate. To validate the models, we applied the analysis on the error/residue. Additionally, we simulated some situations that happen in the real process to verify how the model reacts.

5 Data preprocessing and upload

The primary database comprised about 100 concerned variables from about 900 cells over a period of six years was to be uploaded into the SAP HANA tenant. This primary database has an average sampling frequency of 24 hours. An additional (secondary) database was provided by the smelter containing online data from the control systems, however only the data was collected from only a few cells. The secondary database was used to generate additional variables.

6 Preliminary results

Initially a few cells (24 cells) sharing the same behavior from one potline were chosen to build a thermal and mass balance model of the bath and ledge. To generate the other non-measured variables, we run models using data measure points as initial conditions. Since the mass and thermal balance don't change often, we reduced sampling frequency from one day to 4 hours in average. To generate the data in the 4-hour sampling frequency we used the mathematical models and interpolation techniques for tuning the models. Considering that the cells work similar, no tuning was required in this phase.

In this modeling, about 15 raw variables were used. From these we used another 22 mathematical models to generate other 45 variables. Another 10 variables were added as external input (control). This model should provide the dynamic behavior of all variables involved, so we the model was of type NARX MLP Neural Network, which are simpler yet powerful enough to map the dynamic relation between variables. We

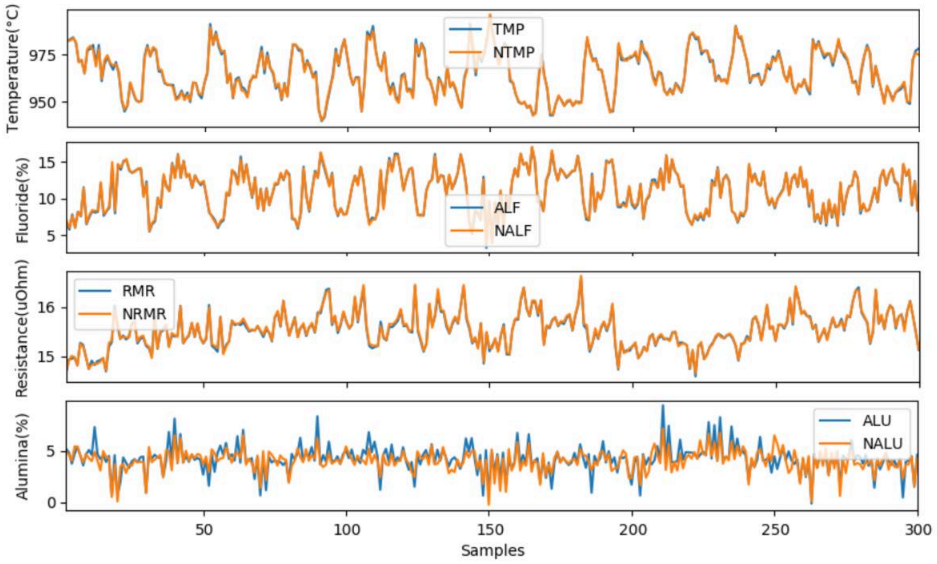


Figure 5: Results of four process variables, where Alumina (4th row) is not directly measured

Table 1: Neural networks with best test performance

Number of hidden Neurons	MSE Test Error
66	0.000 108 0
62	0.000 136 1
71	0.000 158 1
69	0.000 162 3
42	0.000 181 6
61	0.000 212 3
58	0.000 228 5

varied the size of the network from 1 to 100 hidden neurons and used up to three lags. The dataset comprised about 250 000 records, with 85 % for training and 15 % testing. Figure 5 shows the result for each variable (temperature, aluminum fluoride, cell resistance, alumina).

A very good approximation can be obtained in the variables which are directly measured, on the other hand for alumina (4th row), whose values are determined by a mathematical model, the model cannot find a good fit. However, it is known that alumina content changes very often, depending on the feeding frequency among other factors which happen more often in the cell, therefore the neural model tries to reproduce the alumina behavior in average. Table 1 shows the neural networks with the best error measures.

To see how the best model works for a known situation, we simulated a fluoride addition on the cell. It is seen in the chart a severe drop in the temperature because of

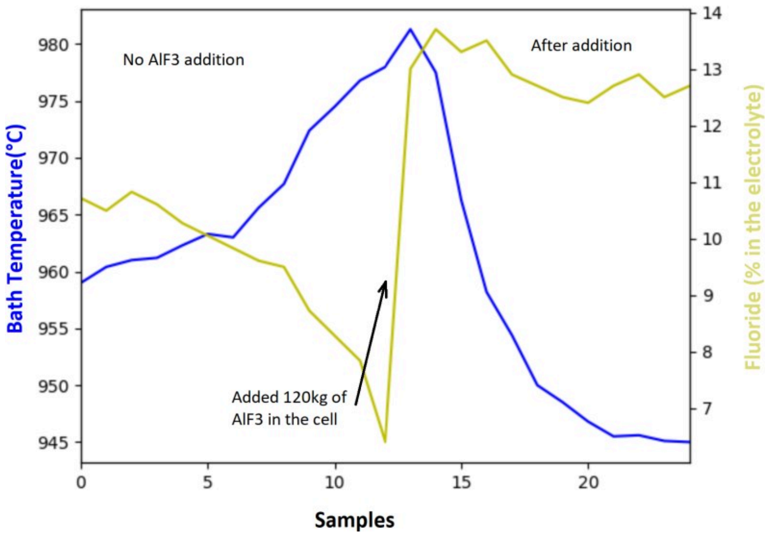


Figure 6: Simulation on a given cell model for a known situation

the growth of excess fluoride in the electrolyte. Moreover, the fluoride content suffers a strong rise immediately, then decaying due to consumption by the production process.

7 Future works

From this preliminary model, we can extend the same approach to other variables and potlines, until all the behaviors of the aluminum reduction cells are mapped in simpler models. Also, other supervised learning algorithms can be tested. It can be viewed in the charts that the model can reproduce the mentioned variables' behavior with reliability, but as they are preliminary results, other conditions need to be tested. The test performed with the fluoride addition action on the model with 66 neurons showed that the expected result is produced, therefore indicating that the approach chosen seems suitable for this kind of work.

References

[1] P. Biedler. "Modeling of an Aluminum Reduction Cell for the Development of a State Estimator". PhD thesis. College of Engineering and Mineral Resources at West Virginia University, 2003.

[2] U. Forsell and L. Ljung. "Closed-loop identification revisited". In: *Journal Automatica IFAC* 35.7 (July 1999), pages 1215–1241. DOI: 10.1016/S0005-1098(99)00022-9.

[3] L. Fortuna, S. Graziani, A. Rizzo, and M. G. Xibilia. *Soft Sensors for Monitoring and Control of Industrial Processes*. 3rd. Springer, 2007.

[4] K. Grjotheim and H. Kvande. *Introduction to Aluminium Electrolysis*. Edited by 2nd. Aluminium Verlag Düsseldorf, 1993.

[5] S. W. Jessen. "Mathematical Modeling of a Hall-Héroult Reduction Cell". Master's thesis. Technical University of Denmark, 2008.

[6] L. Ljung. *System identification: theory for the user*. Edited by T. Kailath and E. Cliffs. 2nd. Prentice Hall, 1999.

[7] A. Wright. "The Dynamic Simulation and Control of Aluminum Smelting Cells". PhD thesis. Newcastle University, 1993.

Architecture-aware Performance and Resilience Engineering for Microservice Architectures

André van Hoorn

University of Stuttgart, Institute of Software Technology, Germany
van.hoorn@informatik.uni-stuttgart.de

This report provides a summary of our project "Architecture-aware Performance and Resilience Engineering for Microservice Architectures" conducted during the HPI Future SOC Lab period spring 2017, as well as ideas for a follow-up project for the upcoming period.

1 Introduction

Modern software engineering paradigms and technologies — such as DevOps [3] and microservices [11] — are gaining more and more attraction in the software and services engineering communities. Of particular interest are quality-of-service concerns, for instance, w. r. t. performance and reliability. While established approaches for classic contexts (i.e., which do not use DevOps and microservices) exist, their adoption to DevOps and microservices requires considerable research efforts [4, 8].

In the recent years, our group has already contributed architecture-aware approaches for performance and reliability, involving a combination of measurement-based and model-based techniques [10, 12, 14]. Recently, we started to investigate how these techniques can be used in DevOps and microservice contexts. We have so far used the Emulab testbed [16], which no longer satisfies our requirements. In order to conduct large-scale experimental evaluations, we need a state-of-the-art computing infrastructure such as the one provided by the HPI Future SOC Lab.

We have requested the HPI Future SOC Lab as an infrastructure for the experimental evaluation of new approaches based on our "CASPA platform for Comparability of Architecture-based Software Performance Engineering Approaches" [5].

In the remainder of this report, we will list the granted Future SOC Lab resources, provide a brief description of our conducted activities as part of the project, and outline next steps.

2 Granted Future SOC Lab Resources

In our proposal, we requested access to a heterogeneous set of computing resources (nodes in the 1,000 Core Clusters, GPU access, cloud (virtual machines)), as well as to storage and database services. In the end, as computing resources, we have been granted (dedicated) root access to three servers: *i.*) 896 GB RAM, 80 cores; *ii.*) 32 GB RAM, 24 cores; *iii.*) 24 GB RAM with an Nvidia Tesla processor. Dedicated access has been given to us due to our expected high resource demands.

3 Project and Findings

The goals of the project were threefold. First, we would like to setup the CASPA platform on the heterogeneous infrastructure. Second, we would like to integrate new components, i.e., software performance engineering approaches, into the CASPA platform. Third, we would like to conduct experiments to evaluate our approaches.

In the remainder of this section, we will provide a brief summary of the microservice application integrated into the platform as well as two new approaches developed and evaluated using this application.

3.1 Sock Shop Microservices Application

As a system under test, we have used the Sock Shop developed by Weaveworks.[1] It represents an e-commerce website and is available under an open-source license. The distributed application has been designed and implemented based on the microservices architectural style [11] and respective state-of-the-art technologies. Each microservice is available as a Docker image, which is a container-based virtualization technology. A recent survey identified the Sock Shop as a candidate for a benchmark application for microservices [1], which was a main reason for choosing it.

We have integrated the Sock Shop application into our CASPA platform, including the integration into Kubernetes. We have used the application as the system under test for the following two projects:

3.2 Detection of Software Performance Antipatterns from Profiler Data

As a follow-up of a GI-Dagstuhl seminar [9] we have developed an approach for detecting and resolving performance antipatterns based on profiler data. In addition to an industrial case study, we have conducted an experimental evaluation in a lab environment. Therefore, we added a profiler agent to the Sock Shop application, executed load tests using JMeter, and applied our detection approach on the resulting profiler results.

3.3 Load Testing of Microservices

BenchFlow[2] [6] is a benchmarking framework that has originally been developed for performance benchmarking of workflow management systems [13]. BenchFlow makes use of container-based virtualization based on Docker and Rancher. We have started to adopt BenchFlow for load testing microservices. We have used the Sock Shop application as the system under test.

[1] https://microservices-demo.github.io/ (last accessed 2017-01-01).

[2] https://github.com/benchflow/ (last accessed 2017-01-01).

4 Next Steps

For the next period, we want to focus particularly on the aspect of continuous performance testing for microservices. We plan the following works:

4.1 Using BenchFlow for Declarative Load Testing of Microservices

We will continue our efforts to use BenchFlow for load testing microservices. We will use one or more sample microservice applications, such as the Sock Shop. During the SOC lab period Spring 2017, we have already started to setup the infrastructure and will be able to continue from this point on. We plan to integrate BenchFlow also into our attempts on Declarative Performance Engineering [15].

4.2 Automatic Extraction and Evolution of BenchFlow Load Test Specifications from APM Data

In our previous work, we have developed the WESSBAS approach [14] for specifying, extracting, and generating workload specificiations for application systems. We will extend the workload extraction from APM data [7] to microservices applications and develop a transformation to BenchFlow load test specifications. The experimental evaluation will be concerned with the accuracy of the extraction and load generation.

4.3 Efficient Performance Testing of Microservices Using Markov Chains

It is not feasible to execute the complete set of available performance tests for each software change as part of the continuous integration automation. Hence, we are interested in selecting and executing only relevant tests. We plan to adopt a seminal approach for efficient selection of performance tests from the telecommunication domain [2] for DevOps and microservices. Particularly, we will use the Markov chain based approach to identify and execute selected load tests using our previously described WESSBAS/BenchFlow approach.

5 Conclusion

We are thankful to the HPI Future SOC Lab for having granted us access to the computing infrastructure. The environment eases the joint work of different organizations on the platform, which has so far been hindered by university-internal access constraints—apart from the fact that an equipment comparable to that of the HPI Future SOC Lab has not been available to us.

Acknowledgment

The presented works are a joint effort being conducted together with many researchers from different organizations, including (in alphabetical order) Alberto Avritzer, Alexander Bran, Andrea Janes, Holger Knoche, Teerat Pitakrat, Barbara Russo, Henning Schulz, Catia Trubiani.

References

[1] C. M. Aderaldo, N. C. Mendonça, C. Pahl, and P. Jamshidi. "Benchmark Requirements for Microservices Architecture Research". In: *Proceedings of the International Workshop on Establishing the Community-Wide Infrastructure for Architecture-Based Software Engineering (ECASE 2017)*. 2017, pages 8–13.

[2] A. Avritzer and E. J. Weyuker. "The Automatic Generation of Load Test Suites and the Assessment of the Resulting Software". In: *IEEE Trans. Softw. Eng.* 21.9 (Sept. 1995), pages 705–716. ISSN: 0098-5589. DOI: 10.1109/32.464549.

[3] L. J. Bass, I. M. Weber, and L. Zhu. *DevOps — A Software Architect's Perspective*. SEI series in software engineering. Addison-Wesley, 2015. ISBN: 978-0-13-404984-7.

[4] A. Brunnert, A. van Hoorn, F. Willnecker, A. Danciu, W. Hasselbring, C. Heger, N. Herbst, P. Jamshidi, R. Jung, J. von Kistowski, A. Koziolek, J. Kroß, S. Spinner, C. Vögele, J. Walter, and A. Wert. *Performance-oriented DevOps: A Research Agenda*. Technical report SPEC-RG-2015-01. SPEC Research Group — DevOps Performance Working Group, Standard Performance Evaluation Corporation (SPEC), Aug. 2015.

[5] T. F. Düllmann, R. Heinrich, A. van Hoorn, T. Pitakrat, J. Walter, and F. Willnecker. "CASPA: A Platform for Comparability of Architecture-based Software Performance Engineering Approaches". In: *Proceedings of the 2017 IEEE International Conference on Software Architecture (ICSA 2017)*. 2017, pages 294–297. DOI: 10.1109/ICSAW.2017.26.

[6] V. Ferme and C. Pautasso. "Integrating Faban with Docker for Performance Benchmarking". In: *Proceedings of the 7th ACM/SPEC on International Conference on Performance Engineering (ICPE 2016)*. Delft, The Netherlands: ACM, 2016, pages 129–130. ISBN: 978-1-4503-4080-9. DOI: 10.1145/2851553.2858676.

[7] C. Heger, A. van Hoorn, M. Mann, and D. Okanović. "Application Performance Management: State of the Art and Challenges for the Future". In: *Proceedings of the 8th ACM/SPEC International Conference on Performance Engineering (ICPE 2017)*. ACM, 2017, pages 429–432. DOI: 10.1145/3030207.3053674.

[8] R. Heinrich, A. van Hoorn, H. Knoche, F. Li, L. E. Lwakatare, C. Pahl, S. Schulte, and J. Wettinger. "Performance Engineering for Microservices: Research Challenges and Directions". In: *Companion of the 8th ACM/SPEC International Con-*

ference on Performance Engineering (*ICPE 2017*). ACM, 2017, pages 223–226. ISBN: 978-1-4503-4899-7. DOI: 10.1145/3053600.3053653.

[9] A. van Hoorn, P. Jamshidi, P. Leitner, and I. Weber, editors. *Report from GI-Dagstuhl Seminar 16394: Software Performance Engineering in the DevOps World*. 2017.

[10] A. van Hoorn. *Model-Driven Online Capacity Management for Component-Based Software Systems*. Kiel Computer Science Series 2014/6. Dissertation, Faculty of Engineering, Kiel University. Kiel, Germany: Department of Computer Science, Kiel University, 2014. ISBN: 978-3-7357-5118-8.

[11] S. Newman. *Building Microservices*. O'Reilly Media, Inc., 2015.

[12] T. Pitakrat, D. Okanović, A. van Hoorn, and L. Grunske. "Hora: Architecture-aware online failure prediction". In: *Journal of Systems and Software* (2017). In press. Online first: https://doi.org/10.1016/j.jss.2017.02.041. ISSN: 0164-1212. DOI: http://dx.doi.org/10.1016/j.jss.2017.02.041.

[13] M. Skouradaki, V. Ferme, C. Pautasso, F. Leymann, and A. van Hoorn. "Micro-Benchmarking BPMN 2.0 Workflow Management Systems with Workflow Patterns". In: *Proceedings of the 28th International Conference on Advanced Information Systems Engineering* (*CAiSE 2016*). LNCS. Springer, 2016, pages 67–82. DOI: 10.1007/978-3-319-39696-5_5.

[14] C. Vögele, A. van Hoorn, E. Schulz, W. Hasselbring, and H. Krcmar. "WESSBAS: Extraction of Probabilistic Workload Specifications for Load Testing and Performance Prediction—A Model-Driven Approach for Session-Based Application Systems". In: *Journal on Software and System Modeling* (*SoSyM*) (2016). In press. Online first: http://dx.doi.org/10.1007/s10270-016-0566-5.

[15] J. Walter, A. van Hoorn, H. Koziolek, D. Okanovic, and S. Kounev. "Asking "What?", Automating the "How?": The Vision of Declarative Performance Engineering". In: *Proceedings of the 7th ACM/SPEC International Conference on Performance Engineering* (*ICPE 2016*). ACM, 2016, pages 91–94.

[16] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar. "An Integrated Experimental Environment for Distributed Systems and Networks". In: *Proc. of the Fifth Symposium on Operating Systems Design and Implementation*. USENIX Association. Dec. 2002, pages 255–270.

Applying Text Mining on Job Offers and Curricula Vitae Using SAP HANA

Skills and Competencies for Industry 4.0

Marlene Knigge, Tamas Neumer, Felix Willnecker, and Helmut Krcmar

Technical University of Munich
Chair for Information Systems
{marlene.knigge,tamas.neumer,felix.willnecker,krcmar}@in.tum.de

Industry 4.0 changes our working environments significantly. Therefore, the goal of our project is to analyze which skills and competencies are required from employees in Industry 4.0. In earlier project phases[1], we implemented an application for analyzing online job offers from the field of Industry 4.0. We applied text analysis and text mining means provided by SAP HANA and the SAP Predictive Analytics Library (PAL) on manually collected job offers. We extracted skills and competencies required for jobs in Industry 4.0 and tried to derive competency profiles for Industry 4.0. In this last project phase, we extended our previous work by providing our application with a different dataset: curricula vitae (CVs) of IT professionals. We adjusted our application for being able to process the new dataset. We then manually compared the results of our CV analysis with our previous results to find out to what extend today's IT professionals are already prepared for working in Industry 4.0.

1 Introduction

"Industrie 4.0" – or Industry 4.0 – or the Industrial Internet enables disruptive solutions by combining known technologies in a new way: the Internet of Things (IoT), Smart Factories, Cyber Physical Systems (CPS), and the increased use of Embedded Systems [3, 4]. This development influences the way we live – and the way we work, by enabling new ways of business value creation, which will result in changing business models and strategies, business processes and a change of the daily working life [4]. Employees need to be prepared to be able to fulfil new requirements [11]: "The requirements for the digitized skilled work will rise because the processes are interconnected and more complex, particularly with reference to the overlap of technical, organizational and social spheres of activity and the work process in the company" [2]. acatech et al. [1] describe in a study regarding competency development, that qualification is one of the essential factors in the Digital Transformation taking place in Germany. For this reason, identifying skills and competencies needed

[1] For further information, please read the project reports of our former projects at HPI Future SOC Lab (Fall 2016 [6], Spring 2017 [5]).

by employees working in an Industry 4.0 environment, becomes an important aspect as it can serve as base for tailoring training and (further) education – or the creation of new job profiles and HR strategies.

Our goal in this project is to extend the knowledge of skills and competencies needed by employees in Industry 4.0. Thus, in our first project phases, we extracted skills and competencies from German online job offers which we manually collected from different job portals using the search term "Industrie 4.0" (German spelling). We analyzed and clustered the results using SAP HANA text analysis and text mining, the SAP Predictive Analytics Library (PAL), and a Python Script.

In our second project phase, we automated the process of extracting the job offers from online portals by implementing a web crawler. We refined our analysis of skill and competency extraction. Moreover, we applied clustering algorithms to our results to discover typical (parts of) (new) job profiles.

In this third project phase, we want to explore to what extend curricula vitae (CVs) of IT professionals already match the requirements of Industry 4.0. Therefore, we want to load the CVs into our existing application and find out, where the application or the analysis of results need adjustment to extract meaningful results from this new data set. Due to the end of the project period, we focus on the technical changes in the implementation, an extended qualitative analysis of the results is not in the scope at this stage.

2 Project Goal

The goal of our project is to extend the knowledge of the analysis of unstructured data from different sources – in this case: German online job offers and CVs. In our two former project phases (cp. [6] and [5]), concentrated of the extraction and analysis of skills and competencies from German job offers that dealt with Industry 4.0. In the current project phase, we want to apply our previous built application on a different dataset: CVs from IT professionals. Our collection contains German and English CVs from IT professionals – it was not chosen by taking into account the search term "Industrie 4.0". In a first step, we want to find out if our application delivers meaningful results when applied on this different kind of data. A second step would be the qualitative analysis of the results from processing the CVs.

We continue using and modifying our application from the former project phases. It is built on SAP HANA and uses text analysis and texted mining means provided by SAP HANA and the SAP Predictive Analysis Library (PAL).

3 Project Design

The first step of this project phase was to gain access to a dataset, which contains CVs from real IT professionals. As this deals with personnel data, this included going through some administrative steps. In the end, we got a dataset for analysis. It may

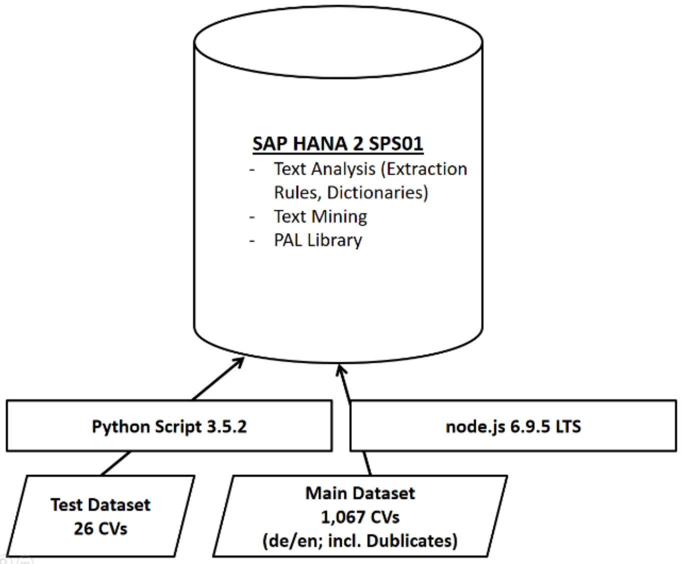


Figure 1: IT architecture project for analyzing CVs (Own illustration)

be possible to get a new dataset if needed, however, it is not possible to update or extend it on a regular base as it is not freely available and updated on the internet or similar.

As a conclusion, for the CVs, an automated extraction from a source is not needed – or not possible. Thus, the automation we implemented in the second project phase is not relevant for this part of the project.

Next, we needed to analyze whether the results our application provided us with were meaningful and discover problems and shortcomings.

While the technical part is working now, our qualitative analysis of the results is still ongoing. In this report, we focus on the technical details of the data processing.

3.1 System Configuration

The Hasso Plattner Institute (HPI) provided us with a SAP HANA 2 SPS01 with 1 TB RAM and 32 CPU cores. Additionally, we used the SAP HANA Predictive Analytics Library (PAL) and the Eclipse-based SAP HANA Studio, version 2.3.10. For uploading job offers, we implemented a script in Python version 3.5.2 using the module "pyhdb" version 0.3.2 as shown in Figure 1. It was possible to reuse our upload script for uploading the CVs.

In this phase, we did not use the previous implemented web crawler, as we got the CV dataset from a project partner as export.

3.2 Dataset

We started with a data sample of 26 CVs in German language for technical testing. These were stored as pdf-files.

Our main dataset comprises 1,067 CVs saved in single files in pdf-, txt-, jpg- or Microsoft Word (docx)-Format. It contains CVs in German and in English language as well as duplicates, e.g., the same file as pdf and docx-file. In the former project phases, we processed html and pdf-files. Therefore, we excluded the jpg-, txt- and docx-files from further analysis. An easy way to include the txt- and docx-files would be to print them as pdf-files. A random check showed, that at least most of the docx-files seem to be duplicates of pdf-files from our dataset. Thus we decided not to convert them for our first analysis. Taking only into account the pdf-files for our further analysis, our main dataset still contains 1,029 files in German and in English language. In this dataset, duplicates may still be comprised.

4 Loading and processing of the Curricula Vitae Dataset

4.1 Loading Dataset

First, we got the data sample that consisted of 26 pdf-files. We had to slightly adjust our implementation to be able to load the new dataset. In the beginning, we were not able to get the dataset into our application as the upload did not work. After several analyses, we finally found that the reason was a data type conflict. After converting all VARCHAR fields in our SAP HANA application to NVARCHAR, the upload and processing of the new data was working. We think that the CV dataset comprised characters or values that did not appear in our job offer dataset. While VARCHAR contains ASCII character strings, NVARCHAR contains Unicode character strings [9]. E.g., VARCHAR(10) comprises 10 single-byte characters; NVRCHAR(10) comprises 10 multi-byte characters [8]. So when using VARCHAR, obviously the data of our CVs did not fit into some fields, whereas it did after changing all VARCHAR fields to NVARCHAR.

After the data was loaded into our SAP HANA application, we were able to execute the analyses implemented in the former project phases without facing any further technical issues.

When we switched to our main dataset, we able to process it without any further issues as well.

4.2 Specifics in the Analysis of Curricula Vitae

In the previous project phases, our focus was on analysing German job offers that are connected to Industry 4.0. Therefore, we implemented custom dictionaries and extraction rules for this context. However, we only implemented them in German language. Our goal was to find out which skills and competencies are needed for working in Industry 4.0. Therefore, these custom dictionaries and extraction rules comprise a whole range of possible skills and competencies – and are not limited to Industry 4.0-specific content. If it was, we would have limited our analysis to only discover what we would expect to be relevant for Industry 4.0. For this reason, our custom dictionaries and extraction rules should be suited to extract skills and competencies from CVs of IT professionals as well. When analysing the documents in SAP HANA, the application detects the language of each document – in our case German or English. We decided to run our analysis on the whole main dataset and to select only the results tagged as German for further analysis later.

We executed our extraction application on the data sample of 26 CVs in German language. We were able to extract 3,085 tokens. These were not only including skills and competencies, but as well metadata such as address, region, and email addresses. However, as we are currently only analyzing on document level, we cannot assign local data to specific sections of the CV documents. A CV document usually contains the address of the author, locations of his education, of former employees or of projects he conducted as customers. This shows us, that without extracting and analysing the structure of a CV and connect it with the extracted values, we have to be very careful with interpreting our results. We always have to think of in which sections of a CV an extracted information such as location data might have been included. E.g., if someone was born in France, but then immediately moved to Germany, he might not even speak one word of French.

Regarding extracted skills and competencies, we should carefully take into account their position in the CVs as well. CVs usually contain information about education, jobs, and, in case of IT professionals, often projects – internal projects or projects at customers. It makes a difference, if someone got in contact with a topic during studies, a short practical, or in a recent project. Or the author may list projects he was part of, e.g. a SAP project, but his task was only to collect the business requirements from the business departments – which does not mean he gained any experiences with SAP.

In conclusion, it is technically possible to analyse CVs with our application which was originally built for analysing job offers. Due to the structure of CVs, we have to be careful when interpreting the results. We would strongly recommend to extract information about the document structure as well and connect it to the extracted skills and competencies for further analysis.

5 Overall Project Results

With this third project phase, we want to conclude the technical part of our project regarding skills and competencies regarding Industry 4.0. However, our limitations and our ideas described in the outlook show areas for further research.

5.1 Limitations

As mentioned before, our application is tailored to analyze German texts as dictionaries and extraction rules are only implemented in German language. It would be interesting to extend it to other languages, e.g., English. This would offer the possibility to compare skill and competency requirements for Industry 4.0 in different regions, e.g., in U.S. of America, United Kingdom, and Germany.

The web crawler implemented in the second project phase is still a node.js standalone application. As it was not subject to our analysis in this last project phase, we did not integrate it into the SAP HANA application yet, although this would be a good solution for avoiding the need to switch systems.

5.2 Results and Outlook

Our application for extracting skills and competencies is working fine on German job offers with a precision around 90 %, which is in line with those of other information extraction systems. The sensitivity of around 70 % is also in line with these. Only the F1-score, which is around 81 % is lower than that of information extraction systems for established problems. Our recommendation is to extend the customer extraction rules and to convert parts of the custom dictionaries into customer extraction rules whenever it is possible to recognize patterns. The further analysis of the results of the CVs may lead to input for new customer extraction rules as well. The Grammatical Role Analysis (GRA), which is only available in English so far, could be helpful to reduce false positives [10].

With clustering of skill and competency requirements from job offers, we got the best results using the Agglomerate Hierarchical Clustering (AHC) algorithm. With this, we discovered 18 clusters which could be combined to competency profiles for Industry 4.0 jobs. From our point of few, the collection of job offers should be extended to put the analysis on a broader basis. The clustering should then be repeated to get a stronger hint on (parts of) competency profiles for Industry 4.0.

Our qualitative analysis of the results of the CV analysis is still ongoing. From a technical perspective, we faced only minor issues in processing CVs instead of job offers. However, we have to be careful when analysing the results as CVs have a different structure than job offers. Other than job offers, CVs do not contain skill and competency requirements for employees. They contain descriptions of skills and analysis of the authors, very often this comprises a detailed description of the education of the author of the CV. Additionally, they have a strong focus on experiences. In IT they often contain descriptions of projects conducted. Thus, we were able to extract a lot of possible skills and competencies of a CV – but so far with no regards to the section of the CV it was extracted from. For a next project, we would recommend to improve the analysis by considering the document structure when analysing the data, as proposed in the master thesis of Tamas Neumer, which was conducted in a parallel project [7]. So far, our CV analysis leads to a first result set that can be compared to the skill and competency requirements extracted from the job offers before to gain a broad overview of existing matches and gaps between supply and demand of skills and competencies.

Last but not least, collecting and analyzing job offers over a longer period may lead to interesting results as well as changes of skill and competency requirements over time may be discovered.

6 Acknowledgments

We would like to thank our student assistant Leon Kevin Birkel for his support.

Moreover, we would like to thank our colleague Loina Prifti, our master student Sebastian Hagn, and our student assistant Tristan Hopf for their support and ideas in earlier project phases.

References

[1] Acatech, Fraunhofer IML, and equeo GmbH. *Kompetenzentwicklungsstudie Industrie 4.0*. 2016.

[2] J. Gebhardt, A. Grimm, and L. Neugebauer. "Developments 4.0 – Propects on future requirements and impacts on work and vocational education". In: *Journal of Technical Education* 3 (2015), pages 117–144.

[3] *Industrie 4.0: Die neue Hightech Strategie – Innovationen für Deutschland*. URL: http://www.hightech-strategie.de/_dpsearch/highlight/searchresult.php?URL=http://www.hightech-strategie.de/de/Industrie-4-0-999.php&QUERY=industrie+4.0 (last accessed 2016-03-09).

[4] H. Kagermann, W. Wahlster, and J. Helbig. *Securing the future of German manufacturing industrie – Recommendations for implementing the strategic initiative INDUSTRIE 4.0. Final report of the Industrie 4.0 Working Group*. Technical report. 2013.

[5] M. Knigge, L. Prifti, S. Hecht, and H. Krcmar. *Follow-Up Project: Automated Text Mining on Job Offers Using SAP HANA: Analyzing Skill and Competency Requirements for Industry 4.0*. Project report for a project in cooperation with the Future SOC Lab at the Hasso-Plattner-Institut. In press.

[6] M. Knigge, L. Prifti, S. Hecht, and H. Krcmar. *Text Mining on Job Offers Using SAP HANA: Analyzing Skill and Competency Requirements for Industry 4.0*. Project report for a project in cooperation with the Future SOC Lab at the Hasso-Plattner-Institut. In press.

[7] T. Neumer. "Efficient Natural Language Processing for Automated Recruiting on the Example of a Software Engineering Talent-Pool". Master's thesis. Technical University of Munich.

[8] SAP Archiv. *In HANA VARCHAR datatype stores unicodes?* URL: https://archive.sap.com/discussions/thread/3570774 (last accessed 2018-03-27).

[9] *SAP optimieren: SAP HANA Reference – Data Types*. URL: http://sap.optimieren.de/hana/hana/html/_csql_data_types.html (last accessed 2018-03-27).

[10] SAP SE. *SAP HANA Text Analysis Language Reference Guide*. URL: https://help.sap.com/doc/2e76b520f80e4fb0b4c91a756f5f51f7/2.0.01/en-US/SAP_HANA_Text_Analysis_Language_Reference_Guide_en.pdf (last accessed 2017-10-24).

[11] *Zukunftsprojekt Industrie 4.0. Digitale Wirtschaft und Gesellschaft*. URL: https://www.bmbf.de/de/zukunftsprojekt-industrie-4-0-848.html (last accessed 2016-03-09).

Hosting of ProteomicsDB at the HPI

Mathias Wilhelm and Bernhard Kuster

Chair of Proteomics and Bioanalytics
Technical University of Munich, Germany
{mathias.wilhelm,kuster}@tum.de

ProteomicsDB[1] is a protein-centric in-memory database for the exploration of large collections of quantitative mass spectrometry-based proteomics data. To date, it contains quantitative data from over 19k LC-MS/MS experiments covering more than 200 tissues, body fluids and cell lines. We extended the data model to enable the storage and integrated visualization of other quantitative omics data. This includes transcriptomics data from e. g. NCBI GEO, protein-protein interaction information from STRING, functional annotations from KEGG, drug-sensitivity/selectivity data from several public sources and reference mass spectra from the ProteomeTools project. The extended functionality transforms ProteomicsDB into a multipurpose resource connecting quantification and meta-data for each protein. The rich user interface helps researchers to navigate all data sources in either a protein-centric or multi-protein-centric manner.

1 Introduction

The large-scale interrogation of biological systems by mass spectrometry based proteomics provides insights into protein abundance, cell type and time dependent expression patterns, post-translational modifications (PTMs) and protein-protein interactions, all of which carry biological information that is best investigated at the protein level. Due to the complexity of proteomic experiments, defining a unified facility to store well-annotated results is challenging. While many efforts to collect and integrate publicly available proteomics datasets exist [2], it is often difficult to retrieve a comprehensive list of identified proteins in a specific biological source or a list of biological sources where a specific protein or post-translational modification is present. Moreover, the lack of integrated meta data and quantitative information often only enables the interaction with identification data, rendering this valuable part of the data inaccessible and futile.

ProteomicsDB [3, 4], a projected initiated by the TU Munich in collaboration with SAP SE (Walldorf and Potsdam), fills this gap and provides access and analytical tools to browse the human proteome. It allows the real-time interactive exploration of large collections of mass spectrometry-based proteomics data. The protein-centric interface not only enables users to quickly access quantification information across all

[1] https://www.ProteomicsDB.org (last accessed 2017-01-01).

experiments stored in ProteomicsDB, but also to view individual peptide evidence. If available, the integrated spectrum viewer automatically selects and presents reference data from synthetic peptides [5] to validate peptide identification events. Using modern web-browser technologies, multiple interactive visualizations are available and enable the real-time exploration of multiple proteomes at the same time. Furthermore, the implementation of an experimental design enables ProteomicsDB to utilize meta-data attached to an experiment, exemplified on dose- and temperature-dependent assay data. This allows the analysis of off- and on-target analysis of drugs as well as the theoretical exploration of combination treatments [1].

ProteomicsDB was hosted by SAP SE and was physically located in Walldorf on a distributed system with 2 nodes (each 1 TB main memory and 80 CPUs; 100 TB storage). The long-term goal is to move ProteomicsDB to the TU Munich. This projects primarily aimed to setup, configure and install a full clone of ProteomicsDB at the HPI to temporary host ProteomicsDB until a permanent solution at the TU Munich was established. Furthermore, this report will highlight three recent advances of ProteomicsDB.

2 Results

2.1 Infrastructure at the HPI

A full backup (~3 TB) of ProteomicsDB from the infrastructure in Walldorf was transferred to the HPI and successfully recovered onto a single node 2 TB HANA instance. Additionally, a separate virtual server was set up to handle requests to R which are used to enable the real-time clustering of proteins and samples within ProteomicsDB. The final infrastructure at the HPI was fully functional and due to the switch from a scale-out to a scale-up system, slight improvement on performance were recognized.

To avoid the transfer of the ProteomicsDB domain to the HPI and subsequently to the TUM, once the final hardware arrived, the infrastructure at the HPI was used as a proxy with a secure connection between the HPI and the TUM.

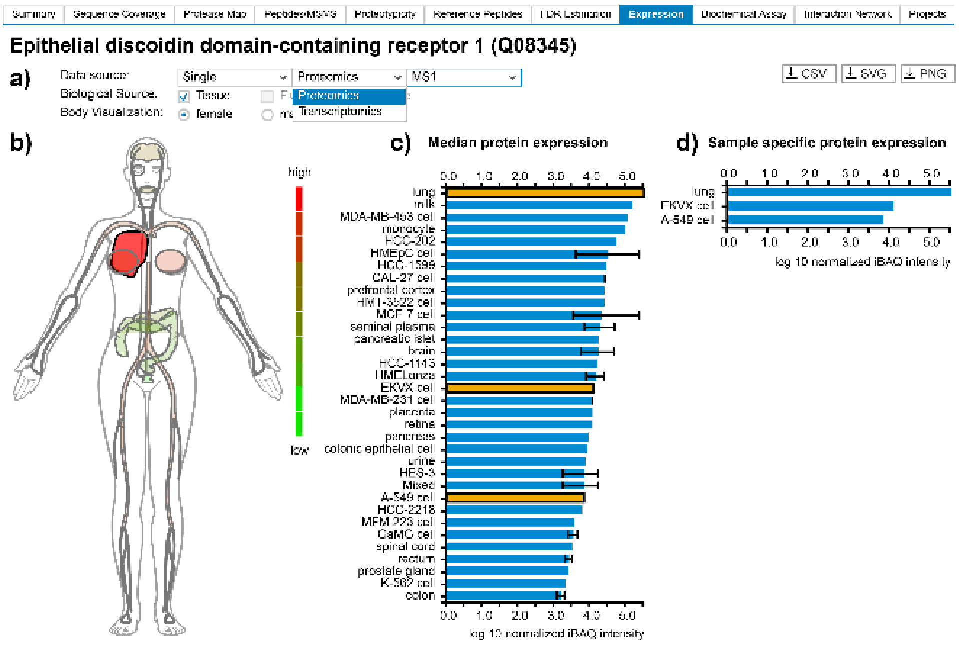
2.2 Advances of the ProteomicsDB data model

The integration of multiple omics, molecular profiling and phenotypic data sources becomes of increasing importance in both academic and health sectors. The goal was to extend the current data model of ProteomicsDB to not only support but also comprehensively integrate such data sources to leverage the biological and biomedical information provided by individual omics levels. While the initial development focused on the presentation of proteomics data, generic implementation of ProteomicsDB also enables the storage and visualization of other omics data types, such as RNA-Seq data.

Besides abundance estimates of proteins, ProteomicsDB now also enables the storage of other omics data. Similar to the proteomics repository, the omics data model organizes samples into experiments and projects. In order to reflect the variety of other omics technologies, this model stores the abundance measure and the measured entity (e.g. transcript) alongside the technology platform and unit provided by the author. Multiple measurements can be attached to a single sample, which facilitates storing e.g. transcript abundances in conjunction with e. g. DNA methylation levels. Figure 1 shows the adjusted visualization of expression values. The user is able to choose, depending on the availability, which omics-level should be highlighted.

The ID conversion functionality – a part of the metadata model – was initially designed to enable the inter-resource conversion of IDs and thus supplements the omics data model. However, due to its generic implementation, it now also serves as an interface to store relations between for example proteins and other proteins, drugs and proteins, as well as a protein's membership in pathways or regulatory networks. All imported entities are automatically clustered into so-called super-nodes to enable efficient and easy navigation of this complex graph of different biological entities (e.g. genes, transcripts, proteins, metabolites) and relations between them (e.g. 'interacts with' or 'activates'). Subsequently, the model was used to store protein e.g. protein-protein interaction data and pathway annotations. The data can be interactively explored (Figure 2) and will enable the integrated analysis of protein-connectivity data with expression information.

The discovery of sensitivity or resistance markers for drugs is becoming a very active field of research and plays a crucial role in evidence based medicine. For most drugs it is not known how and when cells develop a resistance or in some cases do not respond to otherwise highly potent drugs at all. In order to be able to take advantage of the plethora of drug sensitivity information available in the public domain, another triple-store-like model is used to store public drug sensitivity datasets in ProteomicsDB. In addition to the protein-centric data, ProteomicsDB was extended to store phenotypic data from drug sensitivity screens. This allows the association of proteomics data on cell lines included in these screens with their sensitivity/resistance towards thousands of drugs, enabling the discovery of pharmacoproteomic markers of drug response. The drug sensitivity datasets are collections of dose-response experiments measuring the viability (the response) of cancer cell lines as a function of the concentration of a specific drug (the dose) or drug combination. They are annotated with meta-data such as URI and DOI in order to link them to the original publication. ProteomicsDB stores high-level information such as dose-response models and their parameters alongside dose-resolved viability data after normalization, in order to enable the user to estimate the variability of the underlying data. The data model is flexible enough to store experiments with multiple drugs (drug combinations) and can easily be expanded to support e.g. co-culture experiments (cell line combinations) in the future. In cases where raw data are available, this data model allows the comparison of drug sensitivity of the same cell line treated with the same drug across different drug sensitivity datasets, which increases confidence in the data and reduces the number of spurious associations with drug



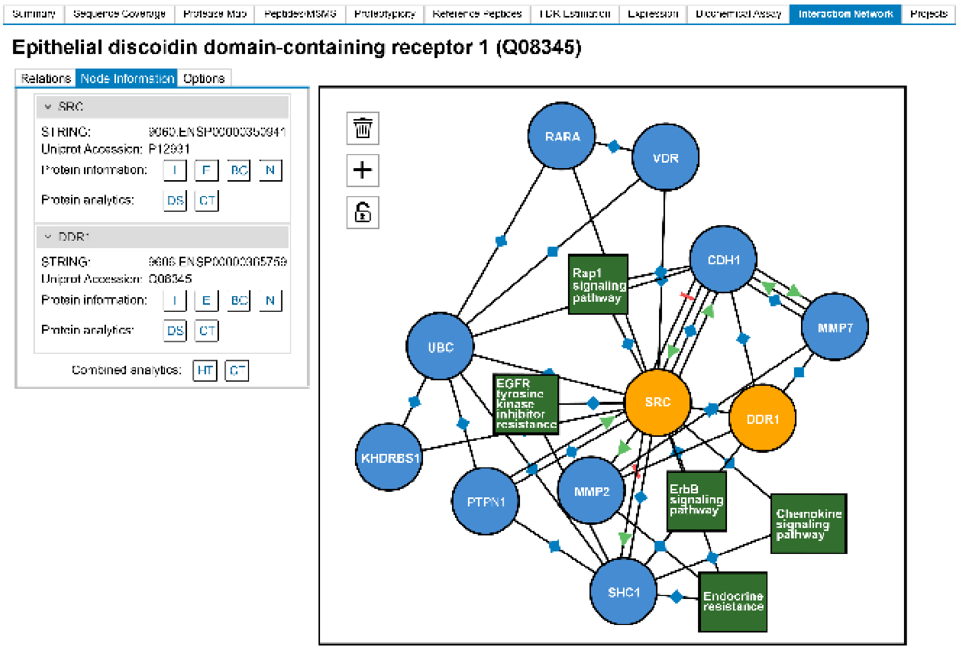
Epithelial discoidin domain-containing receptor 1 (Q08345)

Data source: Single | Proteomics | MS1

Biological Source: ☑ Tissue

Body Visualization: ● female

Figure 1: a) ProteomicsDB can visualize expression data from different omics technologies. b) A heatmap-like bodymap superimposing abundance values of tissues, fluids and cell lines (biological sources) onto their respective tissues of origin. c) A bar chart resolving the expression data of b) on the level of their biological source. If multiple measurements for the same biological source are available, the error bar indicates the lowest and highest abundance observed for the selected protein. The bar chart and the bodymap are linked to each other, enabling the selection of either a tissue of origin in the bodymap (highlighted in dark red) or a biological source in the barchart (highlighted in orange). Here, the lung (high expression of DDR1), was selected in the bodymap, which automatically highlights all corresponding tissues and cell lines in the bar chart (EKVX cell and A-549 cell originated from lung tissue). d) A bar chart visualizing sample-specific abundance values of the sources selected in middle bar chart (highlighted in orange). On click on one of the bars, the corresponding sample preparation protocol can be examined.



Epithelial discoidin domain-containing receptor 1 (Q08345)

SRC

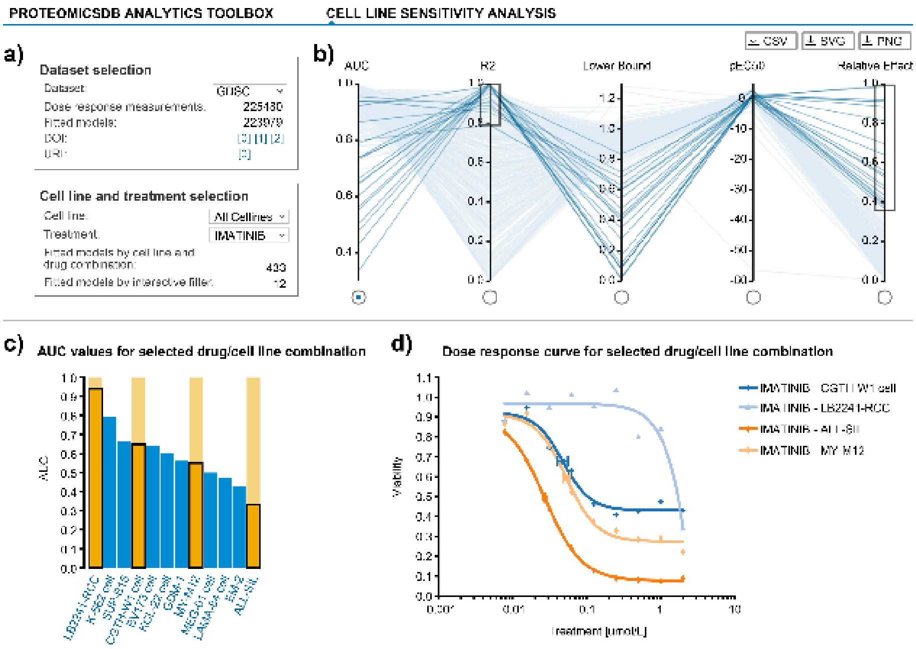
STRING: 9606.ENSP00000350941
Uniprot Accession: P12931
Protein information: I E BC N
Protein analytics: DS CT

DDR1

STRING: 9606.ENSP00000365759
Uniprot Accession: Q08345
Protein information: I E BC N
Protein analytics: DS CT

Combined analytics: HT CT

Figure 2: The interaction graph allows to quickly navigate protein-protein interaction networks and pathway annotations. Proteins and pathways are shown as blue spheres and green rectangles, respectively. Shapes on edges inform about the type of interaction: blue diamonds symbolize known interactions without directionality information as well as functional annotations, red bars indicate inhibitory effects (e.g. SRC inhibits CDH1) and green arrows represent activating effects between two nodes (e.g. SRC activates MMP2). Selected subgraphs and/or proteins (marked in orange) can be directly used for multi-protein centric analyses via "Combined analytics" links (HT: Heatmap; CT: Combination treatment) in the "Node Information" panel on the left, which also enables quick navigation to protein-centric analyses (I: Summary page; E: Expression; BC: Biochemical assay; N: Interaction network; DS: Drug selectivity; CT: Combination treatment).



PROTEOMICSDB ANALYTICS TOOLBOX

CELL LINE SENSITIVITY ANALYSIS

Dataset selection

Dataset: GDSC

Dose response measurements: 225480

Fitted models: 223979

DOI: [3] [1] [2]

URI: [3]

Cell line and treatment selection

Cell line: All Cellines

Treatment: IMATINIB

Fitted models by cell line and drug combination: 433

Fitted models by interactive filter: 12

Figure 3: ProteomicsDB incorporates several publicly available large-scale drug sensitivity screens. a) Each drug sensitivity dataset in ProteomicsDB can be explored in a cell-line- or inhibitor-centric way and general statistics are shown for a given selection. b) Users can interactively filter dose-response models based on multiple parameters such as AUC, R2, lower bound, pEC50 and relative effect (percent decrease in viability over the tested concentration range). c) The distribution of a given parameter is visualized in a bar chart on selection of an axis in b). d) The underlying raw and fitted data can be investigated on click on one or many of the bars (highlighted in orange). The scatter plot highlights the EC50 for the selected cell line:drug pairs. The cell lines CGTH-W1, LB2241-RCC, ALL-SIL and MY-M12 show a clear dose-dependent effect on their viability upon Imatinib treatment. However, their EC50 values vary, highlighting that these cell lines show differential sensitivity/resistance to Imatinib.

sensitivity in pharmacoproteomic studies further down the line. Its visualization and interactive browsing is highlighted in Figure 3.

3 Conclusions

ProteomicsDB was successfully duplicated to the HPI infrastructure and was fully functional. This effort laid the foundation of transferring ProteomicsDB to the Technical University of Munich and led to the success of migrating ProteomicsDB from a little- to big-endian architecture. Furthermore, ProteomicsDB was extended to enable the storage of other omics data, which, as a side product, allowed the storage of additional protein annotations, such as protein:protein interactions. In order to make use of the large body of phenotypic data, a new module enabling the storage of such data was implemented in ProteomicsDB. This will enable the interrogation of drugs and their effects on a systems-biology-wide level since we are now able to integrate multiple omics data with the connectome, phenotypic data and the target space of drugs.

References

[1] S. Klaeger, S. Heinzlmeir, M. Wilhelm, H. Polzer, B. Vick, P.-A. Koenig, M. Reinecke, B. Ruprecht, S. Petzoldt, C. Meng, J. Zecha, K. Reiter, H. Qiao, D. Helm, H. Koch, M. Schoof, G. Canevari, E. Casale, S. R. Depaolini, A. Feuchtinger, Z. Wu, T. Schmidt, L. Rueckert, W. Becker, J. Huenges, A.-K. Garz, B.-O. Gohlke, D. P. Zolg, G. Kayser, T. Vooder, R. Preissner, H. Hahne, N. Tõnisson, K. Kramer, K. Götze, F. Bassermann, J. Schlegl, H.-C. Ehrlich, S. Aiche, A. Walch, P. A. Greif, S. Schneider, E. R. Felder, J. Ruland, G. Médard, I. Jeremias, K. Spiekermann, and B. Kuster. "The target landscape of clinical kinase drugs". In: *Science* 358.6367 (2017). ISSN: 0036-8075. DOI: 10.1126/science.aan4368. eprint: https://science.sciencemag.org/content/358/6367/eaan4368.full.pdf.

[2] Y. Perez-Riverol, E. Alpi, R. Wang, H. Hermjakob, and J. A. Vizcaíno. "Making proteomics data accessible and reusable: Current state of proteomics databases and repositories". In: *Proteomics* 15.5-6 (2015), pages 930–950. DOI: 10.1002/pmic.201400302. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pmic.201400302.

[3] T. Schmidt, P. Samaras, M. Frejno, S. Gessulat, M. Barnert, H. Kienegger, H. Krcmar, J. Schlegl, H.-C. Ehrlich, S. Aiche, B. Kuster, and M. Wilhelm. *ProteomicsDB*. Nov. 2017. DOI: 10.1093/nar/gkx1029. eprint: https://academic.oup.com/nar/article-pdf/46/D1/D1271/23162068/gkx1029.pdf.

[4] M. Wilhelm, J. Schlegl, H. Hahne, A. M. Gholami, M. Lieberenz, M. M. Savitski, E. Ziegler, L. Butzmann, S. Gessulat, H. Marx, T. Mathieson, S. Lemeer, K. Schnatbaum, U. Reimer, H. Wenschuh, M. Mollenhauer, J. Slotta-Huspenina, J.-H. Boese, M. Bantscheff, A. Gerstmair, F. Faerber, and B. Kuster. "Mass-spectrometry-based draft of the human proteome". In: *Nature* 509.7502 (2014), pages 582–587.

[5] D. P. Zolg, M. Wilhelm, K. Schnatbaum, J. Zerweck, T. Knaute, B. Delanghe, D. J. Bailey, S. Gessulat, H.-C. Ehrlich, M. Weininger, P. Yu, J. Schlegl, K. Kramer, T. Schmidt, U. Kusebauch, E. W. Deutsch, R. Aebersold, R. L. Moritz, H. Wenschuh, T. Moehring, S. Aiche, A. Huhmer, U. Reimer, and B. Kuster. "Building ProteomeTools based on a complete synthetic human proteome". In: *Nature Methods* 14.3 (Jan. 2017), pages 259–262. DOI: 10.1038/nmeth.4153.

Aktuelle Technische Berichte des Hasso-Plattner-Instituts

Band	ISBN	Titel	Autoren / Redaktion
129	978-3-86956-465-4	**Technical report : Fall Retreat 2018**	Christoph Meinel, Hasso Plattner, Jürgen Döllner, Mathias Weske, Andreas Polze, Robert Hirschfeld, Felix Naumann, Holger Giese, Patrick Baudisch, Tobias Friedrich, Erwin Böttinger, Christoph Lippert
128	978-3-86956-464-7	**The font engineering platform : collaborative font creation in a self-supporting programming environment**	Tom Beckmann, Justus Hildebrand, Corinna Jaschek, Eva Krebs, Alexander Löser, Marcel Taeumel, Tobias Pape, Lasse Fister, Robert Hirschfeld
127	978-3-86956-463-0	**Metric temporal graph logic over typed attributed graphs : extended version**	Holger Giese, Maria Maximova, Lucas Sakizloglou, Sven Schneider
126	978-3-86956-462-3	**A logic-based incremental approach to graph repair**	Sven Schneider, Leen Lambers, Fernando Orejas
125	978-3-86956-453-1	**Die HPI Schul-Cloud : Roll-Out einer Cloud-Architektur für Schulen in Deutschland**	Christoph Meinel, Jan Renz, Matthias Luderich, Vivien Malyska, Konstantin Kaiser, Arne Oberländer
124	978-3-86956-441-8	**Blockchain : hype or innovation**	Christoph Meinel, Tatiana Gayvoronskaya, Maxim Schnjakin
123	978-3-86956-433-3	**Metric Temporal Graph Logic over Typed Attributed Graphs**	Holger Giese, Maria Maximova, Lucas Sakizloglou, Sven Schneider
122	978-3-86956-432-6	**Proceedings of the Fifth HPI Cloud Symposium "Operating the Cloud" 2017**	Estee van der Walt, Isaac Odun-Ayo, Matthias Bastian, Mohamed Esam Eldin Elsaid
121	978-3-86956-430-2	**Towards version control in object-based systems**	Jakob Reschke, Marcel Taeumel, Tobias Pape, Fabio Niephaus, Robert Hirschfeld
120	978-3-86956-422-7	**Squimera : a live, Smalltalk-based IDE for dynamic programming languages**	Fabio Niephaus, Tim Felgentreff, Robert Hirschfeld
119	978-3-86956-406-7	**k-Inductive invariant Checking for Graph Transformation Systems**	Johannes Dyck, Holger Giese

ISBN 978-3-86956-475-3
ISSN 1613-5652