



UNIVERSIDADE D
COIMBRA

Guilherme José Simões da Cruz

NETWORK HEALTH
INTELLIGENT MVNO OUTAGES UNDERSTANDING
DETECTION AND SELF-HEALING

Dissertation in the context of the Master in Data Science and Engineering,
advised by João Casal, Head of R&D at Truphone, and Professor Hugo Oliveira and
presented to Faculty of Sciences and Technology / Department of Informatics
Engineering.

September of 2022

Faculty of Sciences and Technology
Department of Informatics Engineering

Network Health

Intelligent MVNO Outages Understanding, Detection
and Self-Healing

Guilherme José Simões da Cruz

Dissertation in the context of the Master in Data Science and Engineering,
advised by João Casal, Head of R&D at Truphone, and Prof. Hugo Oliveira and
presented to the Faculty of Sciences and Technology / Department of Informatics
Engineering.

September 2022



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA

Faculdade de Ciências e Tecnologia
Departamento de Engenharia Informática

Network Health

Intelligent MVNO Outages Understanding, Detection
and Self-Healing

Guilherme José Simões da Cruz

Dissertação no âmbito do Mestrado de Engenharia e Ciência dos Dados,
orientada por João Casal, Head of R&D na Truphone, e pelo Professor Doutor Hugo
Oliveira e apresentada ao Departamento de Engenharia Informática da Faculdade de
Ciências e Tecnologia da Universidade de Coimbra.

Setembro 2022



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA

Abstract

In a world where fast and reliable connectivity is ever more a necessity rather than a luxury and where networks grow both in user count and complexity, both architectural and service-wise, there is a necessity for the increased automatization of outage handling, a process that usually requires some degree of human intervention. This document provides a detailed analysis of the state of the art for specific areas of mobile networking relevant for this work, including technical concepts such as Software Defined Networks (SDN) and Network Function Virtualization (NFV) as well as business concepts such as Mobile Virtual Network Operator (MVNO). Furthermore, an analysis of outages in modern connectivity systems is performed as well as a review of self-healing architectures and methodologies. These three main topics are crucial to understand when the goal is to develop a self-healing system in a full MVNO context, since SDN-based infrastructures are inevitably going to be the stage where such a system will function. The in-depth analysis of Truphone's outage history performed during the research period of this dissertation is also presented in this document, including the research for the discovery of the appropriate outage type to target in the self-healing process. During this process, the company was able to discover as well solutions to issues that were causing other outages and consequently one was solved directly as result of these research efforts. The research and development of a self-healing solution for an outage that resulted in failures of calls to Temporary Service Access Numbers (TSAN) is also presented. This solution is comprised of a heuristic-based detection component and a weighted sum model-based healing component. Finally, the document presents the conclusions and suggests future work of the dissertation.

Keywords

Self-Healing, Mobile Virtual Network Operator, Outages, Mobile Networks, Heuristic, Weighted Sum Model, Data Mining, Truphone

Resumo

No mundo de hoje, onde a conectividade rápida e robusta é cada vez mais uma necessidade em vez de um luxo e onde redes crescem tanto em número de utilizadores como em complexidade, tanto em termos de arquitetura como em termos de serviços disponíveis, existe uma necessidade de automatizar o tratamento de falhas, um processo que normalmente requer intervenção humana. Este documento foca-se em fornecer uma análise detalhada do estado da arte para áreas específicas de redes móveis relevantes para este projeto, tais como conceitos técnicos como Software Defined Networks (SDN) e Network Function Virtualization (NFV) e conceitos de negócio como Mobile Virtual Network Operator (MVNO). Além disso, é feita uma análise de falhas em sistemas de conectividade modernos e uma revisão de arquiteturas e metodologias de self-healing. Estes três tópicos principais são cruciais de se entender quando o objetivo do trabalho é desenvolver um sistema de self-healing num contexto full MVNO, visto que infraestruturas baseadas em SDN irão inevitavelmente ser o ambiente onde tal sistema funcionará. A análise aprofundada do histórico de falhas da Truphone realizada durante esta dissertação também é apresentada neste documento, incluindo a pesquisa feita no âmbito de encontrar o tipo de falha mais adequado para ser alvo de self-healing. Durante este process, a empresa foi capaz de descobrir também soluções para problemas que causavam falhas e consequentemente uma falha foi resolvida como efeito dos esforços de pesquisa. A pesquisa e desenvolvimento de uma solução de self-healing para uma falha que resultava em chamadas falhadas para Temporary Service Access Numbers (TSAN) é também apresentada. Esta solução é composta por uma componente de deteção baseada em heurística e uma componente de healing baseada num weighted sum model. Por fim, o documento apresenta as conclusões e sugere de trabalhos futuros da dissertação.

Palavras-Chave

Self-Healing, Mobile Virtual Network Operator, Outages, Mobile Networks, Heuristic, Weighted Sum Model, Data Mining, Truphone

Acknowledgements

I would like to express my gratitude to both of my supervisors, João Casal and Hugo Oliveira, who guided me throughout this project with utmost commitment and availability. I would also like to thank the R&D team, especially José Rosa and Carlos Morgado, and all Truphone members involved in this project for their help and support. A special thanks goes to the members of coolest table in the Department of Informatics bar, with whom I shared two great years. Finally, I would like to thank my family, friends and especially my girlfriend Mariana for always being supportive.

Contents

1	Introduction	1
1.1	Scope	2
1.2	Motivation	2
1.3	Objectives	4
1.4	Approach	4
1.5	Workplan	5
1.5.1	First semester	5
1.5.2	Second semester	6
1.5.3	Work structure at Truphone	8
1.5.4	Deviations from the work plan	9
1.6	Contributions	9
1.7	Structure	10
2	Background	11
2.1	Relevant Trends in Mobile Networks	11
2.1.1	Network Function Virtualization (NFV)	12
2.1.2	Software Defined Networks (SDN)	14
2.1.3	Mobile Virtual Network Operators	16
2.2	Outages in Connectivity Services	21
2.2.1	Outage Causality Studies	21
2.2.2	Outage Detection and Classification	23
2.2.3	Case Studies	26
2.3	Self-Healing Networks	28
2.4	Overview of Data Mining and Machine Learning Concepts	29
2.4.1	Data Mining	30
2.4.2	Machine Learning	33
2.5	Multi-criteria decision making methods	38
2.6	Conclusions	39
3	Related work	41
3.1	Outage Analysis Methods	41
3.1.1	Outage Impact Evaluation	41
3.1.2	Root Cause Analysis (RCA)	44
3.2	Self-healing Systems	45
3.2.1	Outage Detection	45
3.2.2	Outage Diagnosis	51
3.2.3	Outage Compensation	52
3.3	Conclusions	54

4	Understanding Outages at Truphone	55
4.1	Expert knowledge	56
4.1.1	Outage stakeholders' focus group	56
4.1.2	Meeting about Salesforce and how it impacts JIRA data	58
4.1.3	Meetings with the FrontOffice team	60
4.2	Incident Rooms	61
4.3	JIRA dataset	61
4.3.1	Generating the dataset	61
4.3.2	Preprocessing the dataset	62
4.3.3	Truphone's outages and state of the art categories	65
4.4	Results	72
4.4.1	Viable outage types	72
4.4.2	First outage candidate	73
4.4.3	Description of the selected type	74
5	Self-Healing Framework	81
5.1	Data collection and preprocessing	81
5.2	Detection method	82
5.3	Diagnostics method	84
5.4	Healing method	85
5.5	Testing Plan	86
5.5.1	Testing the detection mechanism	86
5.5.2	Testing the healing mechanism	90
5.6	Deployment and Integration Plan	91
5.7	Results	91
6	Conclusion	97
6.1	Future work	99
Appendix A Workplan Timelines		117
Appendix B Key Truphone elements in this dissertation		121
Appendix C Incident Room event flow		123
Appendix D Simulator Data Examples		125
Appendix E Detection Architecture		127
Appendix F Healing Architecture		129

Acronyms

API Application Programming Interface.

ARIMA AutoRegressive Integrated Moving Average.

AS Autonomous System.

AWS Amazon Web Services.

B2B2C Business to Business to Customer.

BGP Border Gateway Protocol.

CAP CAMEL Application Part.

CDN Content Delivery Network.

CDR Call Detail Records.

CNF Cloud-Native Network Functions.

CRISP-DM CRoss-Industry Standard Process for Data Mining.

DDoS Distributed Denial-of-Service.

DNS Domain Name System.

ETSI European Telecommunications Standards Institute.

FCA Financial Conduct Authority.

FTN Forward-to-Number.

GAN Generative Adversarial Networks.

GMM Gaussian Mixture Models.

HLR Home Location Register.

HSS Home Subscriber Server.

IDP Initial Detection Point.

IoT Internet of Things.

ISP Internet Service Provider.

ISUP ISDN User Part.

KNN K-Nearest Neighbours.

KPI Key Performance Indicators.

LDA Latent Dirichlet Allocation.

LOF Local Outlier Factor.

Istm Long Short-Term Memory.

LTE Long-Term Evolution.

MAE Mean Absolute Error.

MCC Mobile Country Code.

MCDM Multi-Criteria Decision Making.

MLN Markov Logic Networks.

MNC Mobile Network Code.

MNO Mobile Network Operator.

MPC Multi-Party Computing.

MSCC Mean Square Contingency Coefficient.

MVNE Mobile Virtual Network Enabler.

MVNO Mobile Virtual Network Operator.

NFV Network Function Virtualization.

NGO Non-Government Organizations.

NLP Natural Language Processing.

ONF Open Networking Foundation.

PCA Principal Component Analysis.

POP Points of Presence.

QoE Quality of Experience.

QoS Quality of Service.

RAN Radio Access Network.

RCA Root Cause Analysis.

RIM Reachability Impact Metrics.

RMSE Root Mean Squared Error.

- ROC** Receiver Operating Characteristic.
- SAW** Simple Additive Weighting.
- SDN** Software Defined Networks.
- SIM** Subscriber Identity Module.
- SIP** Session Initiation Protocol.
- SOM** Self Organizing Maps.
- SON** Self Organizing Networks.
- SSBC** Sum-of-Squares Between Clusters.
- SSE** Sum of Squared Errors.
- SVM** Support Vector Machines.
- TF-IDF** Term Frequency-Inverse Document Frequency.
- TIM** Traffic Impact Metrics.
- TMR** Truphone Mobile Recording.
- TSAN** Temporary Service Access Numbers.
- TSAN** Temporary Service Access Number.
- VNF** Virtualized Network Function.
- VoLTE** Voice over LTE.
- WSM** Weighted Sum Model.

List of Figures

2.1	High Level NFV Framework (Source: ETSI [2013])	13
2.2	SDN high-level architecture overview (Source: Cabaj et al. [2014]) .	15
2.3	SON over SDN/NFV architecture (Source: Moysen and Giupponi [2018])	29
2.4	CRoss-Industry Standard Process for Data Mining (CRISP-DM) process model (Source: Data Science Process Alliance)	30
2.5	Diagram of machine learning types	34
4.1	Impact, Urgency & Priority Matrix Example (Source: bmc.com, Accessed 15/8/2022)	59
4.2	Example of a JIRA ticket (Herbold et al. [2020])	62
4.3	Values for "Problem Root Cause" field in the JIRA ticket dataset . .	67
4.4	Identified root causes	70
4.5	Assigned priorities	71
4.6	Priorities as attributed in the Salesforce dataset	71
4.7	TSAN pre-call flow	74
4.8	Moving average of success rate for Temporary Service Access Number (TSAN) related calls from July 2018 to May 2022 in Spain	79
5.1	Proposed integration plan	92
5.2	Comparison of existing signals and detected successes	93
5.3	Failed retriggers and SIP Invites	93
5.4	Success rate as a percentage	94
A.1	Timeline for semester 1	118
A.2	Timeline for semester 2	119
C.1	Incident Room event flow	124
E.1	Detection mechanism architecture	128
F.1	Healing mechanism architecture	129
F.2	Weighted Sum Model description	130

List of Tables

1.1	Official deadlines	5
2.1	Capability and taxonomy of Mobile Virtual Network Operator (MVNO)s (Source: Balon and Liao [2012])	17
3.1	Perspectives and their advantages and disadvantages	42
3.2	Root Cause Analysis (RCA) works and their target metrics	44
3.3	Summary of outage detection work presented	46
3.4	Summary of outage diagnosis work presented	51
3.5	Summary of outage compensation/recovery work presented	52
4.1	Term Frequency-Inverse Document Frequency (TF-IDF) most significant <i>ngrams</i>	64
4.2	Latent Dirichlet Allocation (LDA) first 20 identified topics and their respective 5 most probable terms	65
4.3	Connection between state of the art categories and categories reported in Truphone	68
4.4	Outage source categories in Truphone data	69
4.5	TSAN-related outages present in the JIRA dataset that occurred in 2022	76
5.1	Confusion matrix of the one hour window size + five minute offset detection tests using real data	94
5.2	Confusion matrix of the one hour window size + one hour offset detection tests using real data	94
5.3	Test results for the detection mechanism using real data	94
5.4	Confusion matrix of the one hour window size + one hour offset detection tests using simulated data	95
5.5	Confusion matrix of the one hour window size + five minute offset detection tests using simulated data	95
5.6	Test results for the detection mechanism using simulated data	95
5.7	Results of the healing mechanism using Z-Score normalization	96
5.8	Results of the healing mechanism using Min-Max scaling	96
B.1	Key Truphone members for the development of the thesis.	122
D.1	Simulated CAP CDR data set	125
D.2	Simulated SIP CDR data set	125

Chapter 1

Introduction

Mobile operator clients demand an increasingly higher quality of service and fault-tolerance of network operators, and this high quality connectivity is ever more a necessity rather than a luxury. Today, this is exacerbated by the advent of more complex network designs and the remotization of work made a reality for many professionals after a worldwide pandemic forced many people to work online from home. And as networks grow in volume and complexity, so do outages¹. This heavily contrasts with a lot of the current methodology that still relies on expert input and manual intervention, which becomes inefficient in dealing with this outage growth and thus becomes obsolete.

A solution to this obsolescence is the automation of outage handling, which may come in the form of self-healing systems, typically considered as one of the three main components of the broader concept of Self Organizing Networks (SON), alongside self-configuration and self-optimization (Feng and Seidel [2008]). Self-configuration in the context of cell networks is the dynamic plug-and-play configuration of newly deployed network elements, whereas self-optimization entails the optimisation of coverage, capacity, handover and interference autonomously. Finally, self-healing deploys features for automatic detection and removal of failures and automatic adjustment of parameters.

This chapter is divided into seven sections: Scope, which describes the context of the dissertation; Motivation, which describes the scientific and business reasons for this dissertation; Objectives, which describes what this dissertation intends to achieve and how that was evaluated; Approach, which describes what was the research approach and methodology chosen; Workplan, which describes how work was divided throughout the dissertation's duration; Contributions, which describes this dissertation's main contributions; and Structure, which briefly describes how this document is structured.

¹2022 Prediction: Internet network outages will continue to get worse before they get better: <https://www.insiderintelligence.com/content/2022-predictions-internet-network-outages-will-continue-worse-before-they-better>, Accessed 1/9/2022

1.1 Scope

This document reports the research and development done in a master's dissertation for the Master's in Data Science and Engineering at the Department of Informatics Engineering of the Faculty of Sciences and Technology of the University of Coimbra. It is supported by Truphone², a GSM Association (GSMA) accredited global mobile network headquartered in London, which provides Mobile Virtual Network Operator (MVNO) services, among others, in 190 countries. Thus, as a provider of network services, Truphone sought to categorize existing outages, select one relevant type of outage and develop and apply to its network a self-healing system capable of recovering from the selected outage type. Such a system proposes a number of benefits for Truphone, namely the reduction of operational costs and the reduction of network downtime, as well as the improvement of quality of experience for both users and businesses.

1.2 Motivation

As many as 4.28 billion people³ routinely rely on an Internet connection for their personal, professional and political lives, making communications an estimated trillion dollar industry⁴. By the time the concept Internet of Things (IoT) was coined by Kevin Ashton in 1999 (Ashton et al. [2009]), only roughly 4% of the world's population was online⁵. In 2017, 46% of people were connected online one way or another and as many as 8.4 billion devices were connected to the Internet⁶. This trend only intensified with the onset of the Coronavirus pandemic, where Internet services saw their usage grow from 40% to 100% of previous years' records and services like Zoom have seen their usage increase tenfold, according to Pandey et al. [2020]. It is clear that connectivity is gaining an increasingly demanding priority in most people's lives.

One of the most problematic issues in any connectivity service are outages, failures that disrupt networks, systems and the capability of a provider to supply its customers with said connectivity. Naturally in such a wide and complex industry, outages are not a rare occurrence. Section 3.1 will explore the nature and effects of outages in greater detail.

Outages come in many shapes and forms and affect businesses and en-

²Truphone's website: <https://www.truphone.com/>, Accessed 20/01/2022

³2020 Mobile internet usage worldwide - statistics & facts: <https://www.statista.com/topics/779/mobile-internet/#dossierKeyfigures>, Accessed 29/11/2021

⁴Mobile E-commerce is up and Poised for Further Growth: <https://www.statista.com/chart/13139/estimated-worldwide-mobile-e-commerce-sales/>, Accessed 27/11/2021

⁵International Telecommunication Union (ITU) World Telecommunication/ICT Indicators Database regarding Internet usage: <https://data.worldbank.org/indicator/IT.NET.USER.ZS?end=2020&start=1999>, Accessed 1/9/2022

⁶Gartner says 8 billion connected things will be in use in 2017: <https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016>, Accessed 1/9/2022

tities of many different types, such as Mobile Network Operator (MNO), social media platforms and cloud computing service providers such as Facebook and Amazon Web Services (AWS), or even the Internet as a whole (Katz-Bassett et al. [2008]). Adding to this, there is no prospect that the frequency of outages will diminish, but rather the opposite, as evidenced by Donegan [2013] and Donegan [2016]. The latter study also estimates that about \$20 billion were spent every year in handling outages in the mobile networking industry. An 2022 OpenGear white paper⁷ that presents an analysis based on a survey of 500 Senior IT decision makers located across North America and Europe also state that 31% of senior IT decision-makers globally said network outages had cost their business over \$1.2 million annually, and 17% stated that this amount was over \$1.6 million.

These outages also translate to increasing costs and expenses to both the companies who suffer them and the users who experience them. In 2021 alone, there were several major outages in connectivity-related services, such as Fastly, Facebook and Cloudflare (see additional information in Section 2.2.3 in Chapter 2) that resulted in millions of dollars in revenue losses. In any industry, this is a heavy price to pay and minimizing it is a great interest to any of the involved parties.

Furthermore, the trend is that with network complexity and traffic increasing over time and with no indication that this trend might subside⁸, outages will only get harder to fix, more frequent and, as a result, impact more people and businesses. In light of this, fixing outages becomes an ever greater task that needs urgent addressing in the networking community.

Naturally, expertise is indispensable in detecting and resolving outages, but automatization of detection and resolution systems can certainly help lowering the necessity of manual intervention in outage handling. For this exact purpose, self-healing began being studied over two decades ago (Elliott and Heile [2000a]) as a mechanism to heal ruptures in wireless networks without the need for an expert to take their time to analyze and identify network information to find an outage.

With the paradigm of networks shifting towards Software Defined Networks (SDN) and Network Function Virtualization (NFV) (see Section 2.1) rather than the classic hardware-centric paradigm, an opportunity arises to leverage these new ways of operating to detect and fix outages. This is one of the foremost motivations for this dissertation, since Truphone, as an MVNO, deals principally with these new concepts in their network infrastructure.

⁷OpenGear's "Measuring The True Cost Of Network Outages: https://www.synnecorp.com/us/westcon/wp-content/uploads/sites/94/2022/05/Email-2-White-paper_MeasuringTheTrueCostOfNetworkOutages_GP_EN_111420-opengear.pdf, Accessed 1/9/2022

⁸2022 Internet and network outages will continue to get worse before they get better:<https://www.emarketer.com/content/2022-predictions-internet-network-outages-will-continue-worse-before-they-better>, Accessed 20/01/2022

1.3 Objectives

The general objective of this project is to study and propose a self-healing solution for a MVNO-specific type of outage, specifically in this case for Truphone. There are a few goals necessary to achieve this, such as:

- Analyzing the state of the art and proposing a categorization of outages in mobile networks.
- Identifying one or more types of outages by analyzing network behavior and prior incidents through existing logs and reports, and relating them to the proposed categories.
- Selecting a relevant outage type for which a self-healing solution can be applied.
- Researching and developing a self-healing solution that can detect, diagnose and mitigate the selected outage type.

This dissertation was successful in delivering a prototype self-healing mechanism. If this prototype were to be further developed and eventually deployed on the network, it expected not only to save Truphone many man hours monthly, but also improve the perceived quality of service for customers that use services affected by the chosen outage. Sections 4.4 and 5.7, as well as chapter 6, describe the impact the research may have on the business in greater detail.

1.4 Approach

The dissertation can be split in two parts, although some of the work was iteratively improved throughout both parts.

The first part encompassed most of the general research to be done, which is studying the state of the art relative to outages, mobile networking and existing self-healing approaches.

This research was used to prepare for the second part of the thesis, which encompassed a deep research of Truphone's outages and network incidents. The deep research was performed through an analysis of outage records (mostly in the form JIRA⁹ tickets) and incident room records, as well as extensive interviewing of various stakeholders in the outage handling methodology in Truphone. In this research, identified outages were categorized as per the state of the art, and a category was chosen to develop a self-healing system using real world data, automating what would otherwise be expert analysis and operation. Finally, this system was tested and evaluated.

⁹What is JIRA?: <https://www.productplan.com/glossary/jira/>, Accessed 29/8/2022

The dissertation followed the CRoss-Industry Standard Process for Data Mining (CRISP-DM) (see section 2.4.1) methodology, with the first semester performing solely Business Understanding while the second semester performed every task of the model, including a deeper Business Understanding, which was done iteratively as expected and defined by the process. See section 1.5 for a more detailed description.

1.5 Workplan

This dissertation presents a project developed jointly between Truphone and the University of Coimbra. The Thesis is divided in two parts, each lasting a semester. The first semester was spent working from the Informatics Engineering Department of the University of Coimbra, being worth 12 European Credit Transfer and Accumulation System (ECTS) credits. The second semester was spent working remotely integrated in Truphone's R&D team and is worth 30 ECTS credits. This accounted for the majority of the 60 ECTS credits that are allocated in a year of study as per the definition of the ECTS¹⁰.

Table 1.1 describes the official deadlines stipulated by the Department of Informatics for the Masters thesis in Data Science and Engineering.

	Description	Date
First semester	Start of dissertation and first meeting	September 20th, 2021
	Delivery of intermediate report	January 24th, 2022
	First public defense	February 2nd, 2022
Second semester	Start of the scholarship at Truphone	February 7th, 2022
	End of the scholarship at Truphone	August 8th, 2022
	Delivery of final report	September 5th, 2022
	Final defense	September 7th, 2022

Table 1.1: Official deadlines

Overall, it was agreed that the work would follow the CRISP-DM process model, which is briefly explained in subsection 2.4.1.

1.5.1 First semester

The first semester kicked-off after the first meeting on September 20th, where both advisors and the student agreed to bi-weekly meetings to discuss work progression. On November 22th, these meetings changed to a weekly basis per the student's request and the advisors' agreement, for a closer follow up of the work progression and more frequent input from the advisors.

¹⁰What is an ECTS?: <https://education.ec.europa.eu/education-levels/higher-education/inclusive-and-connected-higher-education/european-credit-transfer-and-accumulation-system>, Accessed 25/8/2022

The first semester work delves mostly into the Business Understanding section of CRISP-DM, with a few tasks related with organizing the work to be performed. During this semester, the following tasks took place:

- **Defining the research approach:**

The first order of business for this dissertation is establishing the many different tasks to be completed throughout the project. As mentioned previously, this approach is based on the CRISP-DM process model (see Section 2.4.1), following the model's general idea and granularizing it with this dissertation's tasks.

- **Studying the State of the Art:**

After defining a research strategy, it is necessary to research various concepts which are bases for this dissertation. This research begins with mobile networking including topics such as SDN, NFV, MVNO and Truphone itself. This is followed by possible outage classifications and an exploration of the modern day consequences of these outages in businesses and services worldwide, followed by a brief description of the common self-healing framework. Furthermore, relevant data mining and machine learning related concepts are researched as well as multi-criteria decision methods. Extensive research of existing work in outages in connectivity services and Self-healing systems is also performed. The former explores methodology developed for outage impact evaluation, as well as some Root Cause Analysis (RCA) approaches. The latter explores the three components of Self-healing previously defined (detection, diagnosis and compensation/recovery), with greater emphasis to Outage Detection.

- **Defining the second semester's work plan:**

Based on the research done, the second semester's work plan was defined in detail. This task is left for last because it is essentially dependant from the conclusions of the entire research from the first semester.

- **Writing of the thesis document:**

As other tasks get completed, the relevant information is written down in this document, one task per chapter. This document is to be presented in two versions: the first version detailing the first semester's work and the projections for the second semester to be presented in the intermediate defense, and the second version detailing all the work, complete with the state of the art study, approach, results and conclusion, to be presented in the final defense.

A timeline detailing the tasks assigned for the first semester and their corresponding schedule is presented in Figure A.1 of Appendix A.

1.5.2 Second semester

The second semester officially began February 7th, 2022 (according to the University of Coimbra 2021/2022 calendar). This is also the official start date for

Truphone's scholarship which lasted 6 months, ending in August 8th, 2022. The work in this semester encompassed the remaining steps of the CRISP-DM process model, namely Data Understanding, Data Preparation, Modeling and Evaluation (Deployment is dependent on the results produced). There were also changes in Business Understanding which was to be expected, as this model does not imply that steps are immutable once completed for the first time. The projected macro-tasks were divided in three main stages: Outage categorization, Self-Healing and Documentation. The stages and macrotasks were the following:

- **Outage Categorization:**

- **Analysis and categorization of Truphone outages:**

- A detailed research of Truphone's outage history by analysing JIRA records, talking to experts and viewing both concluded and on-going incident rooms to better understand not only what types of outages affect Truphone but also how they were dealt with.

- **Self-Healing:**

- **Selection of a type of outage to handle:**

- A relevant group of outages whose solution can be automated was defined by combining the state of the art research with the Truphone outage analysis. This selection determined how to proceed about developing a self-healing solution, since different outage types require different approaches and in general have different problem statements and nuances about them.

- **Data exploration and infrastructure and protocol research:**

- After an outage type was selected, it was necessary to gain a deeper understanding of the problem to be solved. This meant knowing where relevant data was stored and how it was characterized, as well as understanding base concepts related with the outage type, such as network signalling protocols, what is a Temporary Service Access Number (TSAN) and what is the call signalling flow for a call that uses a TSAN (see Section 4.4.3).

- **Development of the outage detection mechanism:**

- A core part of this dissertation was the development of a self-healing system that is capable of detecting, diagnosing and recovering from outages. The first part of this mechanism dealt with detecting the event of an outage, a process that ended up being strictly heuristic.

- **Development of the outage healing mechanism:**

- The second part of the self-healing system proposed was the healing mechanism, which relied on a Multi-Criteria Decision Making (MCDM) algorithm. In the event of an outage, this mechanism received the identifier of a failing pool and chose the best alternative to replace it in order to mitigate the impact of the outage.

- **Integration of the detection and healing mechanisms:**

The detection and healing mechanisms were developed and tested separately at first. Once it was deemed that they were working as intended, they were tested on an extended set of real world data to gain a more realistic perspective of their performance, as well as simulated data to confer with the experts' definition of the issue.
- **Solution integration/deployment planning:**

Even though it was not realistic to expect that the self-healing mechanism would be deployed during the scholarship's duration, a plan was devised to consider constraints and requirements that the mechanism would need to address if it eventually got to the stage of being deployed on the network.
- **Validation, testing and assessment of the solution:**

An indispensable part of the research and development of the self-healing system was regularly testing it during development to understand if it performed in a satisfactory fashion. This allowed insights not only about the mechanism and the provided data, but also about any assumptions made, including those made by network engineers.
- **Documentation:**
 - **Thesis writing, review and iterative enhancement:**

A critical part of the dissertation is the document itself, which was to be continuously updated and reviewed since the early stages of the scholarship.
 - **Writing the Truphone handbook's entry for this project:**

Truphone possesses an internal handbook which contains information about every project, so it was necessary to write this thesis' entry in that handbook.

Figure A.2 of Appendix A presents the work plan for the second semester's workload, including notable milestones such as the beginning and end of the internship at Truphone in blue, the two prototype deliveries in gray and the thesis submission and defense in orange and red, respectively.

1.5.3 Work structure at Truphone

During the scholarship, the author of this dissertation was working on a full-time regime integrated in Truphone's R&D team. This team uses an agile methodology due to the iterative nature of their work.

As part of this agile methodology, the team has daily meetings to talk about what has been done, what are the ongoing efforts and tasks of each member and what is to be done next. The planning of this agile framework is done for two week sprints, where at the beginning of each the team meets to discuss the success and results of the previous sprint and to plan the tasks to be completed for the following two weeks. In this process, senior members of the R&D

team continuously gave their input and guidance about what needed to be done and provided valuable insights into both the domain of telecommunications and MVNO operations, as well as Truphone's operation, architecture and personnel.

Aside from this, Prof. Hugo Oliveira was also consulted regularly not only to obtain his input regarding data related issues, but also to keep him up to date with important developments of the dissertation.

1.5.4 Deviations from the work plan

There were no major deviations of the presented work plans, and the defined schedule was generally followed without major delays.

1.6 Contributions

This documents describes the research and development of a self-healing solution for an MVNO outage scenario. The achievement of the objectives set resulted in a number of contributions such as the following:

- A state of the art survey regarding outages, mobile networks and self-healing relevant to the proposed solution.
- A categorization of Truphone outages, based on existing literature and on the analysis of Truphone data, and an attempt at aligning this categorization with other categorizations found in the state of the art.
- The well-documented iterative development of a prototype self-healing system in a MVNO context that is able to detect, diagnose and compensate/recover from an outage that affects calls that require use of a TSAN (see Section 4.4.3), with all major design decisions well justified. If applied in production, this system saves man hours¹¹ and increases the customer experience, by recovering from the target outage faster.
- An assessment of the impact that this prototype may have on the business, as well as a suggestion of future work.
- A better and deeper understanding about the TSAN outage at Truphone.
- As a result of the discovery process of the appropriate type of outage to be targeted by the self healing process, other issues that were causing outages were detected, and as a consequence of this an issue was fixed resulting in Truphone saving resources by not having to mitigate the outage it caused.
- A simulator of TSAN-related call records, capable of generating labeled data according to a predefined beta distribution of success rates to create a balanced synthetic dataset.

¹¹Man hour is the cost in terms of cumulative time spent by the people that would manually do the actions to analyze and mitigate the outage event

Furthermore, any and all conclusions that the author deems relevant will be included to further stimulate constructive discussion regarding the topic of self-healing.

1.7 Structure

This document is structured as follows: first, after this chapter which addressed the scope, motivation, objectives, approach, contribution and work plan of this project briefly, Chapter 2 will contextualize the project and explain the necessary concepts and technologies to understand the developed work. Chapter 3 will analyze relevant work that functions as the foundation for the research and development of this project. Chapter 4 details the research of Truphone's outages and presents relevant conclusions, such as a comparison of Truphone's existing outage categories and those proposed by the state of the art research, the definition of what is a viable outage type and the description and justification of the chosen outage type. Chapter 5 presents the research and development process of the self-healing prototype, detailing its architecture, important decisions and their reasoning, data characteristics and their analysis and the results of the prototype. Chapter 6 presents the final thoughts of the document, as well as future work to expand upon the knowledge produced so far.

Chapter 2

Background

This chapter aims to provide context, definition and clarity on the bases of this work. It will begin by giving a comprehensive overview about mobile networking including topics that are particularly relevant for this dissertation, such as Software Defined Networks (SDN), Network Function Virtualization (NFV), Mobile Virtual Network Operator (MVNO) and Truphone itself. This is intended to contextualize the industry and technology where the proposed solution will integrate.

It also describes possible outage classifications and explores the modern day consequences of these outages in businesses and services worldwide, followed by a brief description of the common self-healing framework.

Lastly, a general overview of data mining and machine learning related concepts is followed by a short introduction to multi-criteria decision methods, concluding with the overall conclusions to be taken from this chapter.

2.1 Relevant Trends in Mobile Networks

The evolution from 2G to 4G networks has been mainly driven by the supporting applications, which required new generations of network technologies to incorporate new supporting applications via a redesigning or introducing functionalities (Condoluci and Mahmoodi [2018]). This process requires introducing new hardware elements and in extreme situations the changes to the network could be so disruptive that networks had to be entirely redesigned, leading to the standardization of novel mobile network generations. Not to be ignored are the steep costs in operational and capital expenditures in deploying new network architectures and updating existing ones, respectively.

The bottom line is that the amount of services and applications that need to be supported by networks is growing at faster and faster rates. Continuously redesigning and restandardizing networks is an inefficient way to keep up with surging applications, and is not flexible enough to keep up with evolution of networking dynamics. One great example of this lack of flexibility translating

into slow and difficult changes is the change from IPv4 to IPv6, which is still not even nearly complete even after almost three decades of slowly being phased in.

This evolution comes in the softwarization and virtualization of networks through SDN (Kreutz et al. [2015]) and NFV (Han et al. [2015]). These two technological developments are disruptive game-changers in 5G networks, that are meant to shift the previous hardware-centric paradigm to a software-centric one, providing the much needed flexibility and quick reconfigurability that modern networks require.

Considering that such an evolution creates many new trends, this section focuses on explaining those that were considered relevant for the topic of this dissertation.

2.1.1 Network Function Virtualization (NFV)

Non-virtualized networks implement network functions as a combination of vendor specific software and hardware, often referred to as network nodes or network elements. NFV are intended as the evolution of this model by introducing several improvements, such as decoupling software from hardware, flexible network function deployment and a dynamic way of operating. Figure 2.1 shows the NFV reference architectural framework proposed in ETSI [2013] that identifies the functional blocks and main reference points between such blocks, such as:

- Element Management (EM)
- Virtualized Network Function (VNF)
- NFV Infrastructure (NFVI), including Hardware and virtualized resources and the Virtualization Layer
- Virtualised Infrastructure Manager(s)
- NFV Orchestrator
- VNF Manager(s)
- Service, VNF and Infrastructure Description
- Operations and Business Support Systems (OSS/BSS)

According to ETSI, a VNF is a virtualization of a network function in a legacy non-virtualized network. The EM performs the typical management functionality for one or several VNFs. In NFV, the physical hardware resources include computing, storage and network that provide processing, storage and connectivity to VNFs through the virtualisation layer (e.g. hypervisor). Computing and storage resources are commonly pooled. Network resources are comprised of switching functions, e.g. routers, and wired or wireless links.

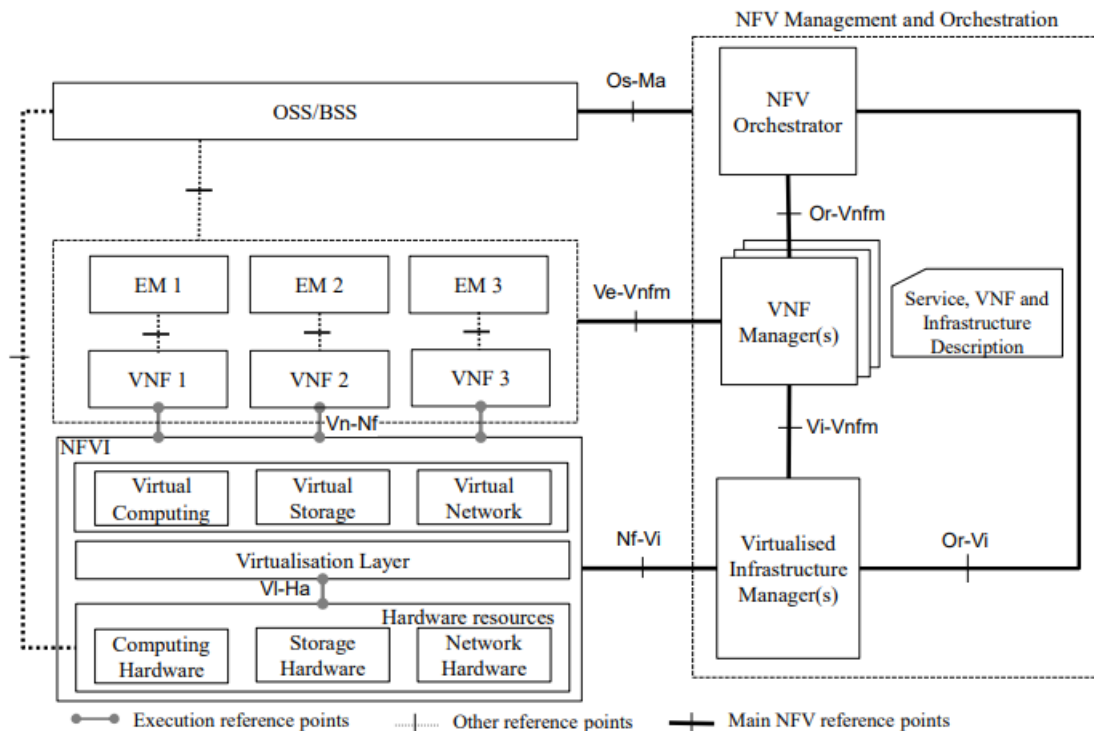


Figure 2.1: High Level NFV Framework (Source: ETSI [2013])

The virtualisation layer abstracts the hardware resources and decouples the VNF software from the underlying hardware, thus ensuring a hardware independent lifecycle for the VNFs. In short, the virtualisation layer is responsible for:

- Abstracting and logically partitioning physical resources, commonly as a hardware abstraction layer.
- Enabling the software that implements the VNF to use the underlying virtualized infrastructure.
- Providing virtualized resources to the VNF, so that the latter can be executed.

Typically, this type of functionality is provided for computing and storage resources in the form of hypervisors and VMs. A VNF is envisioned to be deployed in one or several VMs. The use of hypervisors is one of the typical solutions for the deployment of VNFs. Other solutions to realize VNFs may include software running on top of a non-virtualized server by means of an operating system (OS). To ensure operational transparency, the operation of the VNF is usually independent of its deployment scenario.

Virtualized infrastructure management comprises the functionalities that are used to control and manage the interaction of a VNF with computing, storage and network resources under its authority, as well as their virtualisation. According to the list of hardware resources specified in the architecture, the Virtualized Infrastructure Manager performs resource management and operations,

the latter of which is responsible for Root Cause Analysis (RCA) of performance issues from the NFVI perspective, collection of infrastructure fault information and information for capacity planning, monitoring and optimization. Multiple Virtualised Infrastructure Manager instances may be deployed.

The NFV Orchestrator is in charge of the orchestration and management of NFV infrastructure and software resources, and realizing network services on NFVI. A VNF Manager is responsible for VNF lifecycle management (e.g. instantiation, update, query, scaling, termination). Multiple VNF Managers may be deployed; a VNF Manager may be deployed for each VNF, or a VNF Manager may serve multiple VNFs.

Finally, the Service, VNF and Infrastructure Description dataset provides information regarding the VNF deployment template, VNF Forwarding Graph, service-related information, and NFV infrastructure information models. These templates/descriptors are used internally within NFV Management and Orchestration. The NFV Management and Orchestration functional blocks handle information contained in the templates/descriptors and may expose (subsets of) such information to applicable functional blocks, as needed.

Cloud-Native Network Functions

Cloud-Native Network Functions (CNF) are conceptually similar to VNF. Their main difference is that VNFs imply the use of resource-intensive virtual machines, whereas CNFs make use of cloud technologies such as containerization and orchestration services¹. CNFs are built using microservice architectures operating in the cloud.

These features allow CNF to be far more robust and flexible than VNF, and furthermore allow them to be upgraded, updated and restarted far faster than VNF ever could due to their cloud-native nature. This flexibility and robustness is also well suited to fit the core network needs of 5G-enabled networks, since the standard of these networks employs a Service Based Architecture.

Furthermore, CNF is perfectly suited for the necessary disaggregation of network hardware and software required for standalone 5G.

2.1.2 Software Defined Networks (SDN)

SDN and NFV share much common ground. Both rely on the concept of virtualization that drives the development and deployment of their capabilities. In many ways, SDN and NFV are interdependent, but when deployed together can achieve flexible, agile network infrastructures. But perhaps the greatest difference between SDN and NFV is the fact that whilst NFV deliver the specific functionalities that must be performed at all levels and stages of a network, SDN define the big picture: the type of infrastructure, services and applications available, as well

¹What is a Cloud native Network Function: <https://www.rcrwireless.com/20211025/telco-cloud/what-is-a-cloud-native-network-function>, Accessed 20/01/2022

as a definition of the policies that guide resource usage. Whereas NFV require a hypervisor to coordinate its operation, in SDN a hypervisor orchestrates and controls lower level function². Naturally, this makes both technologies a great fit for each other when deploying software-centric networks, even if they could operate without each other. It is also important to note that while NFV standards are defined and maintained by the European Telecommunications Standards Institute (ETSI), the standardization of SDN is much less evident, although the Open Networking Foundation (ONF) strives to define open industry standards³.

A high-level SDN architecture is presented in Figure 2.2. The ONF defines SDN as such:

In the SDN architecture, the control plane and data plane are decoupled, network intelligence and state are logically centralized, and the underlying network infrastructure is abstracted from the applications

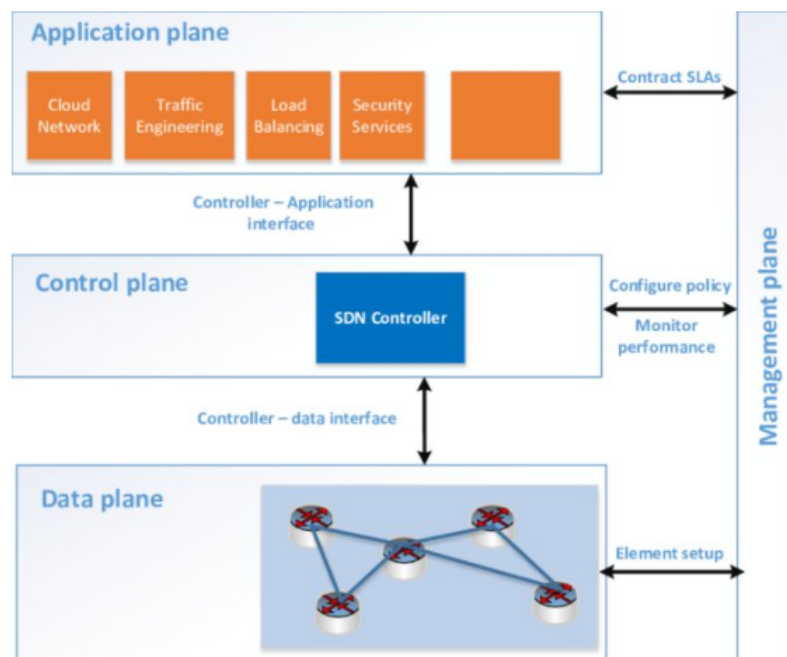


Figure 2.2: SDN high-level architecture overview (Source: Cabaj et al. [2014])

According to Xie et al. [2019], the data plane, otherwise known as infrastructure plane, is the lowest layer in SDN architecture. This plane contains all the forwarding devices, both virtual and physical, including switches. The control plane, on the other hand, is where most of the logic in SDN systems happens. The main component in this plane is a logically centralized controller, which controls the communication between forwarding devices and applications. This works both ways: the controller abstracts and exposes network state information of the

²SDN vs NFV: Understanding their differences, similarities and benefits: <https://blog.equinix.com/blog/2020/03/10/sdn-vs-nfv-understanding-their-differences-similarities-and-benefits/>, Accessed 09/01/2022

³ONF on SDN: <https://opennetworking.org/sdn-definition/>, Accessed 09/01/2022

data plane to the application plane, and constructs custom policies according to requests from the application plane and distributes those policies to forwarding devices. The controller also provides functionalities that the network applications may need, such as shortest path routing, network topology storage, device configuration and state information notifications etc. Finally, the application plane is the highest layer of the SDN architecture, which is composed of business applications. These applications can implement control logic to change network behavior according to its state and to business requirements.

The control plane-data plane connection is called Southbound Interface and the control plane-application plane connection is called Northbound Interface. For the Southbound Interface, the first and most popular standard is OpenFlow (McKeown et al. [2008]), although other less popular proposals exist. There are no de facto standards for Northbound Interfaces, although ONF is trying to define such a standard and common information model.

2.1.3 Mobile Virtual Network Operators

A Mobile Network Operator (MNO) is a telco company provider of wireless voice, text and data communication for its subscribed users. They are independent communication service providers that own the entire telco infrastructure for providing mobile communications between the subscribed mobile users with users in the same or external wireless and wired telco networks. The standard mobile network ecosystem per country is that several mobile operators coexist and each has their own network, infrastructure and international roaming.

Today, however, these ecosystems include several variants of virtual network operators that allow new entrants into the market by reducing the barrier imposed by limited amount of licenses for mobile radio spectrum and the high network investment costs (Balon and Liau [2012]). The main difference between a regular MNO and a MVNO is that whereas the former possesses all layers of the network (from physical to application), the latter does not own a mobile radio license and thus cannot possess the network hardware. Rather, MVNOs use an existing MNO's resources to add value by providing predefined services at a lower cost than a regular MNO. An MVNO usually buys bulk services from an MNO at the network core or access, and this can determine which services the MVNO can offer (Balon and Liau [2012]).

MVNOs can be classified in four distinct categories: service providers, enhanced service providers, light MVNOs and full MVNOs (Balon and Liau [2012]). Table 2.1 provides a general overview of how these categories are separated. Truphone is by definition a full MVNO, because it covers all the features except its own radio spectrum.

The main difference between the full and light categories is that full MVNOs possess their own Mobile Country Code (MCC)\Mobile Network Code (MNC), which allows them to manage their Subscriber Identity Module (SIM) cards via their own Home Subscriber Server (HSS). This allows full MVNOs to more freely switch from one mobile network to another without need to repro-

	Service provider	Enhanced serv. provider	Light MVNO	Full MVNO
Radio spectrum	MNO	MNO	MNO	MNO
Home Subscriber Server (HSS)	MNO	MNO	MNO	MVNO
Mobile Switching Center (MSC)	MNO	MNO	MNO	MVNO
Service platforms	MNO	MNO	MVNO	MVNO
SIM branding	MNO	MVNO	MVNO	MVNO
Billing	MNO or MVNO	MVNO	MVNO	MVNO
Customer care	MNO or MVNO	MNO or MVNO	MVNO	MVNO
Tariff and product development	MNO	MVNO	MVNO	MVNO
Brand visibility to customer	MNO	MVNO	MVNO	MVNO
Customer ownership	MNO	MVNO	MVNO	MVNO

Table 2.1: Capability and taxonomy of MVNOs (Source: Balon and Liau [2012])

gram the customer's SIM cards. The light MVNO, however, does not possess its own MNC but rather reuses the MNO's MCC\MNC and so it is completely dependent on the MNO from which it sources its resources and cannot easily migrate customers from one network to another.

Aside from this, there is also the important question of roaming. Since MVNOs do not possess their own radio access, they cannot offer roaming services to other partners. In fact, a full MVNO can be considered a visiting operator by the source MNO. However, since a full MVNO manages mobility and authentication in its HSS, it can negotiate some of its important roaming routes directly, even though by default these roaming agreements are provided by the MVNO's partners.

There is also another type of business to consider: the Mobile Virtual Network Enabler (MVNE), which acts more or less as a middleman for MNOs and MVNOs by providing core services such as "Voice service control point (SCP), Rating & Billing, B2B customer relationship management (CRM), (Web) Self Care and Order management" (Balon and Liau [2012]). By abstracting IT and technical know-how, it positions itself as a seller of services for smaller MVNOs that may not have the capacity to implement their own core network functions, thus not quite being considered a different type of operator. It can be managed by the source MNO/MVNO or by a 3rd-party.

MVNO advantages

A reasonable question that arises is "why should a company opt to become an MVNO rather than an MNO?", which has a wide plethora of valid responses. According to Lehtikoinen et al. [2014] and Balon and Liau [2012], some of those can be:

- **Operational expenditures**

Any mobile network that seeks to serve a region or country needs to start by applying to get a license to the radio spectrum, which can be an impeding requirement to establishing itself as a service provider since, as mentioned in the Overview section, these licenses are limited in number. But even if

they are granted a license, they still need to invest in a lot of hardware (such as cell towers, GSM devices, significant and stable power supply) that can cover a wide area, translating to a non-trivial starting investment. Furthermore, the network needs to follow many different regulations to keep it secure and needs to have a solid support structure to deal with outages that might arise.

MVNOs do not suffer from nearly as many investment and maintenance issues. Using pre-existing infrastructure avoids getting licenses, hardware and many of the safety measures required of a typical MNO.

- **Focused services**

Since MVNOs do not need to incur in such heavy investment and maintenance costs, they are free to serve more niche markets. This can mean focusing age groups, ethnicities, businesses, etc. For example, Lycamobile is a British MVNO that aims to serve immigrants and expatriates and allow them to make calls to their home country at reduced costs⁴. Another example is Truphone itself, which targets mostly enterprise users that travel often and need a plan that reliably allows them to communicate in several different countries⁵.

- **Higher quality of customer service and care**

As MVNOs are typically more focused on providing services rather than maintaining networks and serve niche markets, customer service and care experiences can take greater priority both in budget and business models. This is an important differentiating factor which acts as a major selling point for customers in any business, including telecommunications (Zhou et al. [2019]).

Truphone

Since this work will be developed using Truphone's data and infrastructure, it is important to understand what is Truphone and what services it offers to its clients for context. Truphone's network technology not only enables the services enumerated in this subsection but also its design is fundamentally guided by these services, so understanding Truphone's business drive is critical to understand the purpose of the technology where the Self-healing system will be acting and receiving information from.

Truphone was a MVNO founded in 2006 in Kent, United Kingdom, by James Tagg, Alexander Straub and Alistair Campbell⁶. It provides several services, namely:

⁴Lycamobile - Fresh Business Thinking: <https://www.lycamobile.com/fresh-business-thinking>, Accessed: 8/12/2021

⁵Truphone takes on the world: <https://www.lightreading.com/mobile/devices-smartphones/truphone-takes-on-the-world/d/d-id/709153>, Accessed: 8/12/2021

⁶Truphone brings internet-rate phone calls to ordinary mobiles at The Cloud's hotspots: https://www.realwire.com/release_detail.asp?ReleaseID=4909, Accessed: 12/11/2021

- **Truphone SIM**

Truphone's first major product is a standard GSM service with patented technology that allows customers to have multiple phone numbers⁷, one for each country where Truphone has MVNO partnerships with local operators. The core network technology deployed throughout these countries is used as local Points of Presence (POP). Depending on where the user is located, a POP will be chosen to reduce the distance that mobile traffic has to travel, reducing latency and improving data speeds and call quality.

Truphone has local rates for each of the 190 countries it has commercial agreements with network operators. These rates are for calls, texts and data. The main target of these services is the business marketplace and Internet of Things (IoT), but personal plans are also available.

Truphone divides its operations in two main classes of countries: Truphone Zone Countries⁸, where Truphone has their core network elements (such as a HSS), and thus can provide phone numbers with these countries' MCC; and Truphone World Countries⁹, where customers can use their minutes, texts and data bundles due to roaming agreements Truphone has in place.

- **Truphone Mobile Recording (TMR)**

TMR is a service additional to the roaming service that allows financial institutions to record mobile calls and SMS, completely compliant with the Financial Conduct Authority (FCA)'s mobile recording legislation and the Dodd Frank Wall Street Reform Act ([Congress, 2010]).

This service is one of Truphone's flagship services, currently used by many of the world's largest financial institutions such as J.P. Morgan & Chase, the Union Bank of Switzerland (UBS) and Citibank¹⁰. In 2021, this service earned the A-Team Innovation Awards 2021, being hailed as the "Most Innovative Operational Resilience/Business Continuity Initiative"¹¹.

This service benefits directly from the proposed self-healing solution, since the solution aims to fix issues that cause the failure TMR calls in specific geographical regions.

- **eSIM platform**

One of Truphone's flagship services, Truphone's eSIM platform allows companies to digitise device connectivity by allowing different profiles to be installed in their users' eSIM enabled devices over the air. Being the first mobile operator to deploy such a platform and being one of the only four

⁷Remote SIM provisioning: <https://docs.things.truphone.com/docs/remote-sim-provisioning>, Accessed: 12/11/2021

⁸Truphone Zone: <https://www.truphone.com/consumer/support/truphone-zone/>, Accessed: 12/12/2021

⁹Truphone World plan: <https://www.truphone.com/support/what-is-truphone-world/>, Accessed: 12/11/2021

¹⁰Truphone for Finance: <https://www.truphone.com/finance/>, Accessed 1/9/2022

¹¹A-Team Innovation Awards 2021: <https://web.truphone.com/about/newsroom/truphone-for-finance-wins-most-innovative-operational-resilience-and-business-continuity-award/>, Accessed 1/9/2022

GSMA eSIM Discovery partners¹², it is clear that this is one of Truphone's growing strengths, especially since eSIM is a fast-growing technology that makes it easier for personal customers to swap plans and business customers to have several different plans for their many needs.¹³

- **MVNO services and global network agreements**

Tying in with the Truphone SIM, Truphone has MVNO agreements with operators in Australia, Germany, Hong Kong, Poland, France, Spain, the Netherlands, United Kingdom and the United States, as well as bilateral roaming agreements with operators in many other countries. Truphone owns its core mobile technology. Because the major distributed points of presence are connected to the core, international mobile traffic can be routed through them, allowing a reduction in physical distance that results in a corresponding reduction call interference and latency and increase in data throughput.

These core components are central to the development of this project, since they are both sources of data and the main targets of action by the proposed self healing system.

- **Truphone as an MVNE**

Mobile services have grown extensively out of the old paradigm where the product was call, text and data packages aimed directly at the consumer. Rather, many different types of business now offer these packages as part of an even more diverse product or service. For instance, a hotel chain may be interested in offering mobile connectivity as part of their room renting deals to allow foreign guests to be able to make calls and text without having to pay roaming fees. Truphone and Manet (a company that offers a concierge app capable of, among other things, providing connectivity via eSIM) struck such a deal in 2021, where Manet uses Truphone's services to provide worldwide connectivity¹⁴.

This type of Business to Business to Customer (B2B2C) business model is the model Truphone seeks for its future rather than direct customer engagement. Truphone offers many services for businesses, notable among which is Truphone Connect that exposes API endpoints to business so they can integrate Truphone's services as an operator (calls, texts and data) in their applications and allowing their customers some sort of connectivity (businesses then decide what they want to offer and purchase Truphone services accordingly)¹⁵.

¹²GSMA's eSIM Discovery partners: <https://www.gsma.com/services/esimdiscovery-smdp/>, Accessed 20/01/2022

¹³Importance of eSIM in mobile networking: <https://www.androidcentral.com/difference-between-sim-and-esim-and-what-future-holds>, Accessed: 20/01/2022

¹⁴Truphone partners with Manet to digitise the hospitality industry: <https://www.truphone.com/about/newsroom/truphone-partners-with-manet-to-digitise-the-hospitality-industry/>, Accessed 20/01/2022

¹⁵Truphone Connect: <https://www.truphone.com/connect/>, Accessed 20/01/2022

2.2 Outages in Connectivity Services

This section analyzes the types of outages that mobile operators face today, their relevance and prevalence, what can cause them, and how they relate to one another. It is useful to keep in mind that outages evolve over time and what may have been a slight issue a few years ago may be today's major headache. The same goes for future outage types. As stated by Aceto et al. [2018]:

Network operators need systems and systematic approaches to detect in (near) real-time ongoing Internet outages. These early warning systems are essential to limit the impact of these type of events.

It is also worth considering that the tools and approaches that a MVNO may have access to is a smaller subset of the tools and approaches to solve issues that a MNO possesses, due to the abstraction from the physical layer. Conversely, a MNO may not have the flexibility that an MVNO possesses when it comes to picking and choosing the best access points for their customers. Not only this, but both MVNOs and MNOs usually have many different functions acting as separate services on offer, which can be affected by network outages as well as possess their own failure points.

2.2.1 Outage Causality Studies

There are two major types of failure in mobile networks: outages and degradations (or, alternatively, full and partial outages as per Asghar et al. [2018]). In full outages, one or more network components/services completely stop working and do not or cannot execute their intended functions. These can be, for example, servers that crashed, antennae that stopped responding or physical links that were broken. In partial outages (or service degradations), components do not completely stop working but instead work in a sub-optimal fashion. For instance, if a certain server is expected to handle 1GB/s of data but is instead only handling 100MB/s of data (and so working at 10% expected capacity) and this is causing a noticeable bottleneck, its function can be considered degraded. Both full and partial outages can be caused by numerous different reasons. Furthermore, outages can be classified as outages with known and unknown causes. Known outages are outages that have previously occurred and whose causes have been identified and understood by experts, whereas unknown outages are outages that have yet to be detected or identified, which can mean anomalous behavior in the network that have not yet been studied by domain experts.

An extensive high-level analysis of outages and degradations in mobile networks can be found in Donegan [2016], where 54 author-picked correspondents that work in telecom answer a form regarding outages and degradations in telecom. This survey concludes that degradations are expectedly far more common than full outages, and problems usually arise at service-level (one service is affected) rather than network-level (all services are affected). Naturally, the greatest cause for service degradation is network congestion, whereas network

failures, physical link failures and application/server issues are close seconds, third and fourth, respectively, while network failures are widely recognized by the correspondents as the outages that take the longest to identify and fix. A growing concern is cyberattacks, which mark a significant rise over their consideration in the 2013 survey, but were not considered at the time as one of the major causes of outages or degradations. Today, cyberattacks are much more prevalent and a cause for greater concern, according to Steinberger et al. [2015]. Most operators recognize some difficulty in identifying an outage's root cause, and there is a strong reliance on network monitoring tools, call center statistics and network traffic analysis. The domains most affected by outages and degradations are the Radio Access Network (RAN), the transport network and the routing layer, which might be worthwhile noting are usually beyond a full MVNO's reach. However, there is a large impact on the voice and data components of the mobile core as well as Internet access, SMS infrastructure and network enabler components (such as Home Location Register (HLR)/HSS), which are the operating layers of MVNOs.

A cloud service-specific survey regarding outages was conducted by Li et al. [2013]. This survey collected 112 cloud service outage events from 2007 to 2012, and analyzed details about these outages such as which were the providers affected, duration of the outages, geographical data and root causes. Perhaps the most interesting takeaways from this survey are that 79% of outages resulted in upwards of one hour of downtime, and more than half of these (43% of the total) lasted between one to six hours. 16% of outages had a downtime of over 24 hours. Even if services only suffered on average short outages, the sum of downtimes could be several dozens of hours, as was the case for Gmail that had an overall downtime of about 72 hours in 2008, even though it had an average of 10 to 15 minutes per outage. It is also worth mentioning that Gmail's userbase had a growth of over 428% between 2012 and 2018¹⁶, so the number of outages is also expected to have grown. Furthermore, this survey concluded that the most prevalent root cause of outages was system issues such as software bugs or updates, downed servers, recent changes to hardware, human misoperation or misconfiguration, hacks and equipment overloads. This is closely followed by power outages to server facilities and routing or network issues like hardware issues regarding core devices, infrastructure or routing devices, and software bugs, communication errors and HTTP errors. The outage root cause classification scheme used by this survey can be found in subsection 2.2.2.

Regarding security issues, Steinberger et al. [2015] details the analysis of a survey with 42 respondents hailing from mostly European Internet Service Provider (ISP) and other network operators about collaborative attack mitigation and response. Some key takeaways are that Distributed Denial-of-Service (DDoS) attacks, which are perceived as the most popular type of attack and affect 58% of the correspondents once or twice a month on average, affect mostly mid-sized networks whereas transport networks are less affected by this type of attack. The authors provide two possible reasons for these facts: one is that attackers may predominantly target end users and another is that these attacks may

¹⁶Active Gmail Users according to Statista.com: <https://www.statista.com/statistics/432390/active-gmail-users/>, Accessed 1/9/2022

not have a great enough impact in high-speed networks for service providers to notice them. The survey results also indicate that 49% of correspondents identify less than ten security incidents per month via automatic systems such as Intrusion Detection Systems, management systems and security information, whereas 20% detect over 500 incidents per month. However, 74% of correspondents indicate that only 10% of detected incidents turn out to be real security events that need to be handled.

2.2.2 Outage Detection and Classification

Aceto et al. [2018] offer a comprehensive survey of Internet outage-related literature. An interesting product of this survey is the taxonomy of outages as they divide causality in three major aspects: origin (natural vs. human caused), intentionality (accidental vs. intentional) and types of disruption (physical vs. logical or mixed). Another interesting conclusion is that it is essential to dissect and analyze individual episodes of outages to understand how networks react globally and locally to disruptive events.

Furthermore, Donegan [2016] split outages into 9 different categories:

- Network Failures
- Physical link Failures
- Network congestion/overload
- Customer device issues
- Configuration issues
- Application/server issues
- Malicious damage
- Network/service enablers
- Cybersecurity-related attack

Li et al. [2013] use the following classification scheme:

- **Power Outages:**
 - Direct Power cut: Power to the facility has been cut
 - Hardware: Hardware¹⁷ responsible for the distribution of power have been disabled
 - Human Mistake: Human input causes a power disruption

¹⁷Breakers, Bus Ducts, Cables, Electrical Grounds, Power Distribution Units, Programmable Logic Controllers or Transfer Switches

- Natural Disaster: Damage caused by a natural disaster causes a power disruption
- Uninterruptible Power Supply (UPS) Issue: When main power supplies fail, UPS systems automatically intervene to bridge the gap while the auxiliary power source is not engaged, and a failure in such systems can cause a power outage
- Vehicle Accident: a vehicle accident has disrupted power supply
- **Routing/Network Issues:**
 - DNS Error: Errors accessing a Domain Name Server (DNS), which can be due to bad domains, loss of connection, among others
 - Hardware: Networking hardware such as a Core Device, Infrastructure or a Routing Device has been disabled, causing an outage
 - Human Mistake: Misoperation or Misconfiguration
 - Request Flood: A high volume of incoming network traffic can generate an outage
 - Software: Bug, Communication Error or HTTP Error
- **(Other) System Issues:**
 - Database Error: A database or its data has been corrupted or made inaccessible, causing an outage
 - Hack: DDoS Attack or Virus
 - Hardware: Chiller Failure (failure of a cooling system), Recent Change or Server Down
 - Human Mistake: Misoperation or Misconfiguration
 - Overload: Memory Leak or Request Flood
 - Software: Bug or Recent Change
 - Storage Error: A failure in a storage device can cause an outage
- **Third-Party Outages**

Gunawi et al. [2016] analyze 1237 news and post-mortem reports that detail 597 unplanned outages from 2009 to 2015 in Cloud Services, and use the following categories as root causes for outages:

- Unknown
- Upgrade
- Networking Failures
- Bugs
- Misconfigurations

- Load
- Cross-Service Dependencies
- Power Outages
- Security Attacks
- Human Errors
- Storage Failures
- Miscellaneous Server/Hardware Failures
- External and Natural Disasters

The Study Group for Security, Reliability, and Performance for Software Defined and Virtualized Ecosystems (SRPSDVE) also made a study of the Internet, Cloud and SDN/NFV outage categories and possible approaches¹⁸ in which they also promptly mention that there is no standardized classification methodology for Cloud Service outages and present categories from several different sources, namely the ATIS NSRC (Alliance for Telecommunications Industry Solutions' Network Reliability Steering Committee) and QuEST Forum's TL-9000 categories for equipment in outage classification/analysis and Ofcom and ENISA's outage classifications. The former two classifications are not as relevant for the purpose of this work because in MVNO contexts the failing equipment is not relevant in most cases. The latter two classifications provide a few valuable categories, many of which coincide with categories presented in other works, such as:

- Hardware Failure
- Third Party
- Transmission Failure
- Power Outage/Surge
- Software Failure/Bug
- Human Error
- Malicious Attack (physical or cyber)
- Natural Disaster
- Congestion
- Process Failure

¹⁸Classification of Internet, Cloud and SDN/NFV Service Outages: Observations and Possible Options/Approaches, https://grouper.ieee.org/groups/srpsdv/meetings/2015%20April/Makris_SDN_NFV%20Outage%20Classification%20Options_08April2015.pdf, Accessed 25/3/2022

- Planned Work
- Miscellaneous Server/Hardware Failures

These works have some common classifications and where they differ they complement each other, and provide an interesting benchmark to compare Tru-
phone's outages. It is important to note, however, that most of them are relatively high level and inside the same categories can exist problems with very distinct solutions.

2.2.3 Case Studies

Anyone working with information systems, whether mobile networks, software products or something else, that rely on an infrastructure with many moving parts has without doubt faced outages of one type or another. Network outages have been an issue for quite some time. Katz-Bassett et al. [2008] encountered reachability issues¹⁹ with over 10.000 different prefixes²⁰ out of a total pool of over 110.000 probed prefixes. A total of 60% of the connection issues they had lasted for over 2 hours, with close to 20% lasting over 10 hours.

Similarly, the author of Heavy Reading's 2016 Mobile Network Outages & Service Degradations survey (Donegan [2016]) estimates that about \$20 billion were spent every year in dealing with these serious outages and degradations.

But costs do not come in the shape of direct spending alone. Amazon itself theoretically lost \$66.240/minute (totalling almost \$2 million) in sales revenue in service outage in 2013²¹.

In June 15, 2020, T-Mobile suffered a 12-hour plus outage that led to congestion across their 4G, 3G and 2G networks, resulting in the failure of 23.000 911 (The United States of America's emergency services number) calls. The severity of this fact led to the FCC fining T-Mobile \$19.5 million in November, 2021.²²

In October 4th, 2021, Facebook faced one such outage during a routine maintenance operation in their data centers. During this operation, a command that assesses the global backbone capacity accidentally disconnected all of Facebook's data centers. Since their Domain Name System (DNS) servers were designed to withdraw their Border Gateway Protocol (BGP) routes if they could not

¹⁹Situations where a prefix is unreachable or very slow to respond from several different access points.

²⁰One can think of a prefix as the IPv6 equivalent of IPv4 netmasks: a portion of the IP address used to identify addresses that belong to the same network (Subnet masks (IPv4) and prefixes (IPv6): <https://www.ibm.com/docs/en/ts3500-tape-library?topic=formats-subnet-masks-ipv4-prefixes-ipv6>, Accessed 8/10/2021)

²¹Amazon.com Goes Down, Loses \$66,240 Per Minute: <https://www.forbes.com/sites/kellyclay/2013/08/19/amazon-com-goes-down-loses-66240-per-minute/?sh=4d6f2fce495c>, Accessed 04/01/2022

²²T-Mobile to pay nearly \$20 million after outage leads to thousands of 911 calls failing: <https://abcnews.go.com/Business/mobile-pay-20-million-outage-leads-thousands-911/story?id=81369531>, Accessed 06/01/2022

connect to the data centers, even though the DNS servers ran in a separate network, resulting in the complete disconnection of Facebook from the Internet. This outage resulted in a 4.9% decrease in the company's market valuation that same day.²³

Similarly, Cloudflare (which handles approximately 18% of all web traffic according to Mathenge [2021]) experienced a major network bottleneck in July 17th, 2020. Resulting from a configuration change made on their backbone network, all BGP traffic was rerouted to another backbone router in Atlanta which quickly became overwhelmed, resulting in heavy congestion. The solution was dropping the affected router and rerouting traffic to other routers. This is similar to Akamai's edge DNS outage, that also stemmed from a configuration update that triggered a bug. The issue in this case was addressed by an update rollback.

In June 8th 2021, Fastly (a cloud Content Delivery Network (CDN) provider that counts among its customers companies like Amazon, Twitch, Reddit, CNN and even the UK government) suffered a major global outage that resulted in as many as 85% of all routing requests to its network returning status code 503 "Service Unavailable", even though the outage was reportedly identified after one minute and resolved in 49 minutes. The cause behind this outage was a bug introduced in a prior deployment that was triggered by specific customer configurations under specific circumstances. These configurations and circumstances materialized June 8th.

In November 9th 2021, Comcast Xfinity suffered a massive outage that disconnected TV and Internet from users in several regions of the United States of America, including the cities of Chicago, New York, New Jersey and Philadelphia.²⁴ The root cause has not yet been disclosed as of since, as the only communication from Comcast was that:

Some customers are experiencing intermittent service interruptions as a result of a network issue

and:

Our teams are actively working to bring impacted customers back online, as we continue to investigate.

²³Facebook Engineering blog post explaining the October 4th 2021 outage: <https://engineering.fb.com/2021/10/05/networking-traffic/outage-details/>, Accessed 27/12/2022

²⁴Comcast Xfinity mass outage: <https://www.nbcchicago.com/news/local/xfinity-responds-after-mass-comcast-outages-reported-in-chicago-area-illinois/2678418/>, Accessed 20/01/2022

2.3 Self-Healing Networks

To understand self-healing, one must first understand the wider concept of Self Organizing Networks (SON)²⁵. SON is divided in three components: Self-configuration, self-optimization and self-healing. Self-configuration is related to solutions that autonomously configure network elements to allow plug and play. Self-optimization is about systems capable of autonomously reconfiguring themselves to optimize performance according to operator needs. Self-healing is dedicated to solutions capable of detecting, diagnosing and compensating/recovering from performance issues such as outages and degradations. Aside from these three components, there is another component that serves more or less as a controller for the SON Functions executed by each of the core components called Self-coordination. These intelligent network functions have been subject of research for quite some time in mobile cellular networks, having been first introduced by Elliott and Heile [2000b]. This section will only explore concepts related to self-healing.

The 3rd Generation Partnership Project (3GPP)²⁶ is a partnership between several standard development organizations in telecommunications and since Release 8 has been iteratively defining SON-related standards. Perhaps most notably, in Release 10 concepts such as Cell Outage Detection and Cell Outage Compensation have been defined. The main defined use cases are: Self-recovery of a Network Element's software, Self Healing of hardware failures in the Network Elements and Cell Outage Management (Moysen and Giupponi [2018]).

Ramiro and Hamied [2011] state that any Self-healing framework should be comprised of three main stages: Outage Detection, Outage Diagnosis and Outage Cure. They also state that there are occasions in which the Self-healing functionalities can only indicate the existence and perhaps the root cause of a problem, the necessity of mandatory human intervention prevents automatic resolution of said problem from being a possibility.

Asghar et al. [2018] classify the Outage Cure stage as Outage Compensation and consider Outage Recovery as a possible additional stage, but in this project we consider Outage Compensation and Outage Recovery to be parts of the same stage.

The Outage Detection stage is about continuously monitoring the network in search of possible outages. Once an outage is detected, data regarding its source, cause, type, etc. will be collected in the Outage Diagnosis step. Finally, a compensation/recovery action is selected based on the outage's information in the Outage Compensation/Recovery stage.

According to Quattrociochi et al. [2014], self-healing strategies in cellular networks aim at maintaining network connectivity by assuming the possibility of creating new communication channels among the nodes of the networks. This

²⁵3GPP Standards for Self-Organizing Networks: <https://www.3gpp.org/technologies/keywords-acronyms/105-son>, Accessed 12/01/2022

²⁶About 3GPP: <https://www.3gpp.org/about-3gpp>, Accessed: 20/01/2022

is not the case in cellular networks, where the possibility to create new links is normally not available in the short timespan where self-healing systems are supposed to react, since links are physical and creating new ones requires both time and investments. This restriction is lightened in software-centric networks.

The relative novelty of SDN has not given a chance for researchers to uniformize and standardize self-healing in that particular architecture, even though a considerable amount of solutions have been proposed (see Chapter 3). Many of these solutions, whether in cellular networks or in SDN-enabled networks, involve machine learning techniques, which are quickly proving to be a considerable asset in improving accuracy and effectiveness in self-healing mechanisms, even in cellular contexts (Ali-Tolppa et al. [2018]). SELFNET²⁷ is a notable approach that proposes a SON (which self-healing is a part of) over SDN/NFV architecture, as can be seen in Figure 2.3.

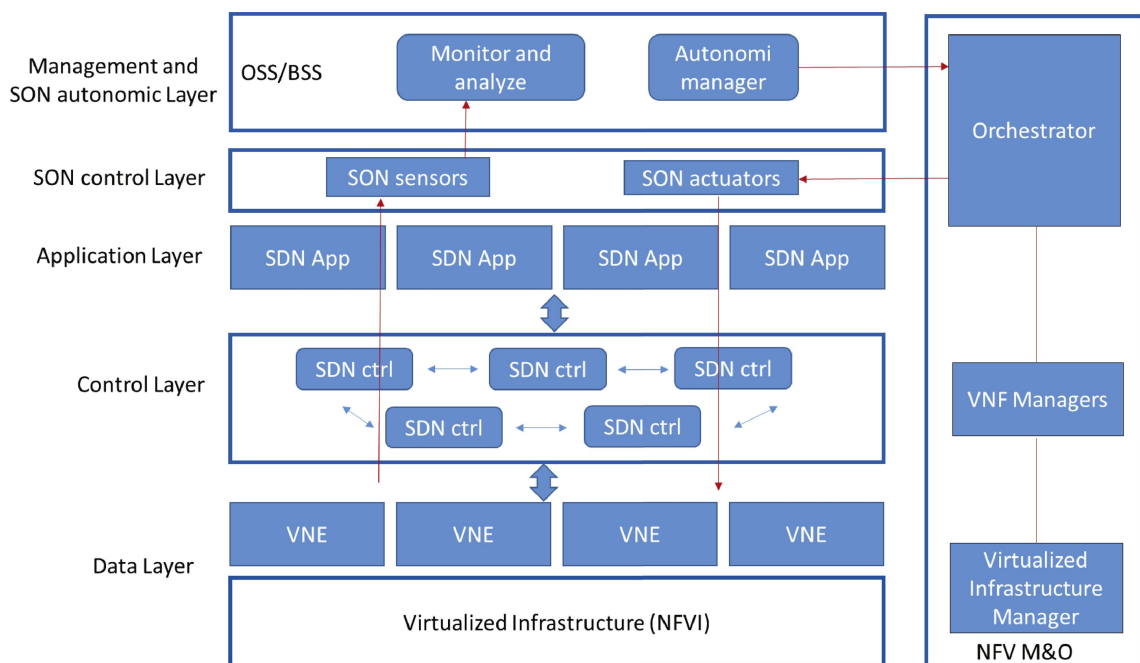


Figure 2.3: SON over SDN/NFV architecture (Source: Moysen and Giupponi [2018])

2.4 Overview of Data Mining and Machine Learning Concepts

Understanding and developing self-healing solutions for an MVNO is as reliant on knowledge about networks as it is on knowledge about data science concepts such as data mining and machine learning, so this section provides some fundamental concepts on those subjects.

²⁷SELFNET: <https://5g-ppp.eu/selfnet/>, Accessed 21/01/2022

2.4.1 Data Mining

Data mining is the process of extracting or discovering patterns in large data sets or databases through methods such as statistical algorithms, machine learning, text analytics, text mining, time series analysis or other analytical methods. It is generally thought as a field that involves both computer science and statistics. A critical part of handling data is also the extraction and preprocessing of the data itself, which is considered Data Preparation as per the Cross-Industry Standard Process for Data Mining (CRISP-DM).

The CRISP-DM (Wirth and Hipp [2000]) is a staple process model in the data science community that describes how a data science project should go about (see Figure 2.4).

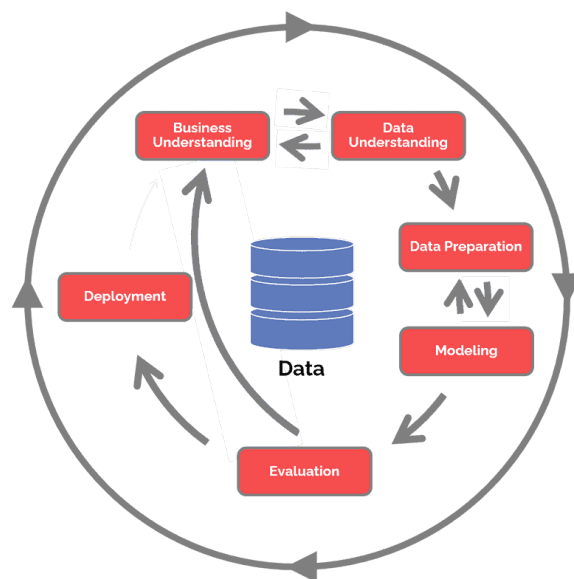


Figure 2.4: CRISP-DM process model (Source: Data Science Process Alliance)

It entails six steps²⁸, each attempting to answer one or more relevant questions:

- **Business Understanding:** What does the business need?
- **Data Understanding:** What data do we have/need? Is it clean?
- **Data Preparation:** How do we organize the data for modeling?
- **Modeling:** What modeling techniques should we apply?
- **Evaluation:** Which model best meets the business objectives?
- **Deployment:** How do stakeholders access the results?

²⁸CRISP-DM: <https://www.datascience-pm.com/crisp-dm-2/>, Accessed 10/01/2022

The whole process' viability is a compound of each step's effectiveness. For example, if Business Understanding is poor, then the notion of what data is needed may be incorrect. If the data is not well understood, then the preparation step, even if it is still done correctly, will generate data that may not be good enough to create a model that can solve the issue at hand. If the resulting model is not good enough, then evaluation will not produce good results. Choosing the wrong evaluation metrics can also mean that even if the evaluation produces good results (Hossin and Sulaiman [2015]), those results will not translate well to the real world and the deployed model will not produce any significant improvement and can actually be harmful for the business.

As per Munson [2012], data preparation and engineering occupies the majority of the time budget in most projects. Data engineering can be defined as the task of making raw data usable for data science tasks. This can be seen as the Data Understanding and Data Preparation steps, and can mean doing statistical analyses of data, combining data from different sources, combining existing variables into new synthetic variables, etc.

The Modeling step usually resorts to some sort of machine learning approach, such as heuristic, statistical, deep learning, etc.

Data Engineering

Real world data is rarely ready for use out of the box. For this reason, Data Engineering is crucial for the vast majority of projects that deal with real world data, as previously mentioned. Some tasks part of the Data Engineering/Preparation process are the discretization of data, removing outliers and noise from the data, integration of data from various sources, dealing with incomplete data and transformation of data to comparable dynamic ranges (viz. normalization). They are all always necessary, but often a combination of these tasks is necessary.

An important part of Data Preparation is normalizing data, as stated by Singh and Singh [2020]. It deeply impacts the performance in classification, before even training models. An alternative to normalizing data is scaling it. There are some popular normalization and scaling methods, such as min-max scaling and z-score normalization.

Min-Max Scaling scales data between the maximum and minimum values of the feature's range, usually between 0 and 1 or -1 and 1 ([Eesa and Arabo, 2017]), although the formula can be adapted to scale between any x and y bounds. While it is considered good for when the data's distribution is not Gaussian and/or the values are fixed in a known range (e.g. pixel color values are always bound between [0,255]), this method of scaling is not sensitive to outliers. The formula for the min-max scaler is presented in equation 2.1.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} (Bound_{max} - Bound_{min}) + Bound_{min} \quad (2.1)$$

Unlike Min-Max Scaling, Z-Score normalization, also known as Standard

Scaling, is sensitive to outliers (Kappal et al. [2019]). Instead of considering the maximum and minimum bounds of the feature range, it takes the mean and standard deviation into account. The normalized feature is not bounded between any finite values. This method of normalization assumes that the data follows a Gaussian distribution, and thus does not deal well with any skewness or kurtosis. The formula for the z-score is presented in equation 2.2.

$$x_{normal} = \frac{x - \mu}{\sigma} \quad (2.2)$$

Text Mining

A sub-task of Data Mining is Text Mining, which deals with extracting information and insights from unstructured texts. This sub-task is interesting in this project because the JIRA dataset (see Section 4.3) is comprised of a large corpus of JIRA tickets with much information as long-form text. As per Tandel et al. [2019], some of the steps in text mining are:

- Converting unstructured data from different types of sources into structured data
- Preprocessing and cleaning text to uniformize it and remove anomalies
- Reducing words to their base form to recognize word roots and extract patterns from the structured data, by either
 - Stemming, which is the process of reducing inflected words to their stem, i.e.: "issue", "issuing" or "issued" are reduced to "issu".
 - Lemmatizing, which is much like stemming but returns a meaningful word/representation rather than a stem, i.e.: "issue", "issuing" or "issued" are reduced to "issue".
- Cluster textual documents to learn underlying patterns

In particular, unsupervised clustering techniques such as Latent Dirichlet Allocation (LDA) or Term Frequency-Inverse Document Frequency (TF-IDF) can greatly help extracting patterns and understanding underlying trends in text data (Jelodar et al. [2019], Qaiser and Ali [2018]).

LDA (Blei et al. [2003]) is a generative probabilistic model of a corpus. It represents topics by word probabilities, where the words with highest probabilities in each topic usually represent the main themes of that topic. Documents can then be represented as mixtures of these topics. It is one of the most commonly used topic modelling methods.

According to (Jelodar et al. [2019]), LDA assumes that each document can be represented as a probabilistic distribution over latent topics, and that topic distribution in all documents share a common Dirichlet prior. Given a corpus D consisting of M documents, with document d having N_d words ($d \in 1, \dots, M$), the way the corpus D is modelled is:

- Choose a multinomial distribution ϕ_t for topic t ($t \in \{1, \dots, T\}$) from a Dirichlet distribution with parameter β .
- Choose a multinomial distribution θ_d for document d ($d \in \{1, \dots, M\}$) from a Dirichlet distribution with parameter α .
- For a word w_n ($n \in \{1, \dots, N_d\}$) in document d ,
 - Select a topic z_n from θ_d .
 - Select a word w_n from ϕ_{z_n} .
- The probability of observed data D is calculated and obtained from a corpus as:

$$p(D|\alpha, \beta) = \prod_{d=1}^M \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d \quad (2.3)$$

LDA, in essence, allows the unsupervised clustering of many text documents into their most relevant topics, which are groups of words and their corresponding probabilities, represented as probabilities in Dirichlet distributions.

TF-IDF is a combination of two concepts: Term Frequency (Luhn [1957]) and Inverse Document Frequency (Jones [1972]). Term Frequency is defined as

$$TF(t) = \frac{t_d}{T_d} \quad (2.4)$$

where t_d is the number of times term t appears in document d and T_d is the total count of terms in document d . Inverse Document Frequency is defined as

$$IDF(t) = \log_e \left(\frac{D_T}{D_t} \right) \quad (2.5)$$

where D_T is the total number of documents and D_t is the total number of documents with term t in them. The TF-IDF weight of any given term t for a corpus of documents D_T is the product of TF and IDF. It is used for tasks such as stop-word filtering, but can also be used for more advanced tasks such as unsupervised clustering of documents (Bafna et al. [2016]).

A component of Text Mining is Natural Language Processing (NLP), which concerns more with linguistic understanding and structure of text data rather than the application of data mining techniques such as clustering and classification.

2.4.2 Machine Learning

Machine learning is a field in artificial intelligence and computer science which studies algorithms capable of adapting to a problem without need of explicit instructions. As previously mentioned, the Modelling component of CRISP-DM is very reliant on machine learning, but it can also be used in the data preparation

step. Machine learning is a major field in data science as a whole. Raschka [2015] provides good fundamentals about this field. There are four main methods of machine learning²⁹: supervised, unsupervised, semi-supervised and reinforcement. Figure 2.5³⁰ shows the different machine learning types and what distinguishes them.

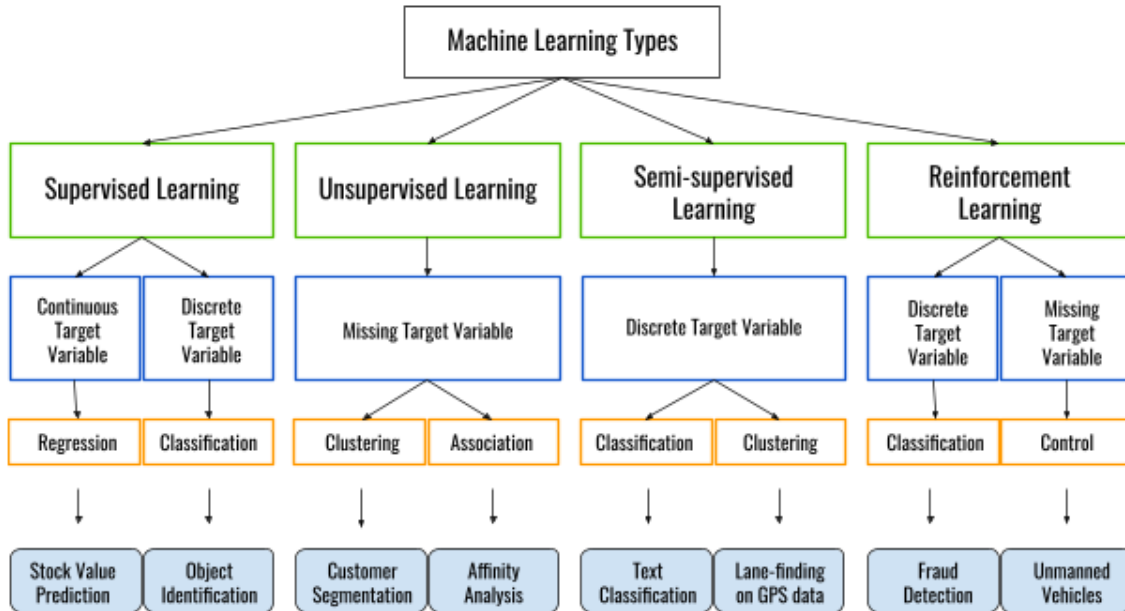


Figure 2.5: Diagram of machine learning types

²⁹Types of Machine Learning Algorithms You Should Know, <https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861>, Accessed 11/01/2022

³⁰Adapted from <https://en.proft.me/2015/12/24/types-machine-learning-algorithms/>, Accessed 23/01/2022

Supervised Learning

Supervised learning entails generating a model by fitting a machine learning algorithm to a training dataset. The trained models are designated predictive. Then this model is tested in a testing data set to evaluate the quality of the trained model. It is important to note that the training and testing data sets should not share any data points (or samples), as this would produce an incorrect assessment of the trained model.

There are two types of problems in supervised learning: classification and regression. Classification algorithms try to find the best separation of data by their labels – or classes – and predict the labels of new entries according to the criteria they have been taught. In this method, the labels are qualitative and discrete, and the classification of a data point relies solely on their attributes. Regression, on the other hand, is about learning to make predictions of continuous values rather than specific discrete values. It is commonly used in time series prediction, for example, to make projections, such as sales revenues, weather forecasts, etc.

There is a large variety of classifiers with many different advantages and disadvantages for both classification and regression tasks. Some of the most popular supervised classification techniques are Decision tree classifiers, K-Nearest Neighbours (KNN), Neural Networks (such as Multi-Layer Perceptrons or Recurrent Neural Networks), Naive Bayes Classifiers, and Support Vector Machines (SVM) Classifiers (Narayanan et al. [2017]). Some of the most popular supervised regression techniques are Linear Regression, SVM Regression, Decision Tree Regression, Random Forest Regression and Lasso Regression (Choudhary and Giney [2017]).

There are many ways of validating models. One of these methods hold-out validation, which is the method of splitting the dataset in train and test. A validation data set can also be used to estimate the model's quality when tuning its hyperparameters, by instead splitting the dataset in train, validation and test.

Another popular method of validating a model's quality is through k -fold cross-validation (Wong [2015]), where the existing data set is divided in k different subsets. Then, the model is fitted on $k-1$ subsets and tested on the remaining subset. This is done until all combinations of the subsets are exhausted and the final result is k different evaluations of the model. There is a particularly interesting case of cross-validation called leave-one-out cross-validation, where each fold is a single sample of the data set. This method is very good at maximizing training data while minimizing bias, although it can be very computationally taxing due to the necessity of training a model for every sample in the data set. Cross-validation has the advantage of producing more insightful evaluations whereas hold-out is less computationally expensive and thus better for large data sets.

These validation methods are usually not used together. Since the objective of these methods is to evaluate whether the model learned the signal without learning irrelevant noise, combining the two methods results in a reduced amount of training data and is a redundant method of evaluating models.

Finally, a crucial part in validating a model is choosing which evaluation metric to use. One of the most popular evaluation metrics for classification problems is the F1-score, which is calculated by combining precision and recall (another two evaluation metrics) through their harmonic mean. Considering TP as "True Positives", TN as "True Negatives", FP as "False Positives" and FN as "False Negatives", the formula for precision in a two class problem is

$$Precision = \frac{TP}{TP + FP} \quad (2.6)$$

whereas recall's formula, also in a two class problem, is

$$Recall = \frac{TP}{TP + FN} \quad (2.7)$$

Finally, the F1-Score is calculated as

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (2.8)$$

Chicco and Jurman [2020] consider the Mean Square Contingency Coefficient (MSCC), more commonly known as the Matthew's Correlation Coefficient, more reliable than the F1-Score, because it is not sensitive to which class is considered positive and which class is considered negative, unlike F1. The MSCC is calculated for a two class problem as

$$MCC = \frac{TP * TN - FN * FP}{\sqrt{(TN + FP) * (FN + TP) * (TN + TP) * (FN + FP)}} \quad (2.9)$$

Both are extensible to a multi-class problem, but the MSCC involves less calculations because F1 requires averaging the F1 scores for each class, which results in the need to calculate an F1 score for each class. MSCC does not require this.

Another relatively common metric is accuracy, calculated as

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (2.10)$$

However, this metric is very sensitive to class imbalance. For instance, if there are 990 positive samples, all of which are classified correctly by the model, and 10 negative samples, all classified incorrectly by the model, accuracy is $\frac{990+0}{990+10+0+0} = 0.99$, which is arguably a good accuracy even though the model naively classifies everything as positive. This metric is also easily extensible for a multi-class problem.

For regression problems, there are a few popular metrics such as the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) (Chai and Draxler [2014]). The MAE is calculated as

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n} \quad (2.11)$$

where y_i is the prediction, x_i is the true value and n is the total number of data points. The RMSE is calculated as

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (x_i - \hat{x}_i)^2}{N}} \quad (2.12)$$

where N is the number of non-missing data points, x_i is the true time series value and \hat{x}_i is the predicted time series value.

Unsupervised learning

Unsupervised learning, unlike supervised learning, is when models are trained with unlabeled data. The trained models are designated descriptive. Rather than being taught by humans about the connections in the supplied that set, unsupervised learning seeks to learn unseen patterns that humans may not have found. They are very commonly used for pattern detection and descriptive modelling.

The main types of learning algorithms in this category are Clustering and Association rule learning. Clustering algorithms such as K-Means and DBSCAN are often used for anomaly detection (Ahmed et al. [2016], Agrawal and Agrawal [2015]). Deep learning methods are also very popular for more complex clustering applications (Guo et al. [2017]).

According to Palacio-Niño and Berzal [2019], validating models created with unsupervised learning techniques can be made using internal/unsupervised validation or external/supervised validation. Internal validation does not rely on external data and relies on methods that strictly use information about the clusters such as evaluating cohesion and separation, either using the Sum of Squared Errors (SSE) and Sum-of-Squares Between Clusters (SSBC) metrics or through methods such as the silhouette coefficient (Rousseeuw [1987]), through similarity matrices, or through hierarchical methods such as the cophenetic correlation coefficient (Saraçlı et al. [2013]). External validation methods incorporate additional information such as external class labels for the training examples and can be metrics used for supervised techniques, such as F1-score, precision and recall, peer-to-peer correlation metrics such as Jaccard index or Rand index, and information theory techniques such as entropy and mutual Information.

Since the majority of unsupervised learning problems do not have access to external labelled data, internal validation methods are more popular than external validation methods.

Semi-supervised learning

As labeling data can be an expensive and time-consuming task, semi-supervised (also known as weak supervised) methods serve as a medium between unsupervised and supervised learning methods. In this method, training data sets are

composed by a small amount of labelled combined with a larger amount of unlabeled data. Two major avenues of semi-supervised learning are transductive learning and inductive learning.

Transductive learning seeks to attribute the correct labels to the unlabeled data, whereas inductive learning seeks to learn correct mapping between the labelled and unlabelled data.

A popular semi-supervised learning method is using Generative Adversarial Networks (GAN) (Creswell et al. [2018]), which begins with a small subset of trained data, and learns to generate artificial data points.

Reinforcement learning

In reinforcement learning, models, commonly agent-based, try to find the optimal way to reach a goal or accomplish a certain task. Any step an agent takes can be in the direction of said goal – in which case it may receive a positive reward – or in a non optimal direction – in which case it receives a negative reward. These methods rely on positive and/or negative rewards to optimise the performance of the model, and are by nature iterative processes, working on the basis that the more iterations the model goes through, the better notion of what it must do to optimize results. Usually, agents do not attempt to just choose the best immediate action, but instead attempt to maximize long-term gains.

Algorithms of this type are usually divided into model-free ([Çalışır and Pehlivanoglu, 2019]) and model-based (Moerland et al. [2020]) approaches. Whereas model-based approaches use experience to create a model which will be the basis of their actions, model-free approaches attempt to learn state/action values and/or policies without modelling the world they find themselves in. The latter methods are less efficient than the former, since they start with more assumptions about the agent's world, but are less resource intensive to train.

2.5 Multi-criteria decision making methods

Multi-Criteria Decision Making (MCDM) methods are used to make decisions when multiple criteria (or objectives) need to be considered together in order to rank or choose between alternatives. It is a sub-discipline of operations research, which in turn is considered to be a sub-field of mathematical sciences. The purpose of these methods is to support decision-makers facing such problems. Many problems do not possess a single optimal solution, and instead the best choice depends on decision-makers' preferences.

Aruldoss et al. [2013] presents some popular MCDM methods. The method most relevant for this dissertation is the Weighted Sum Model (WSM), also known as Simple Additive Weighting (SAW) (Yang [2014]). Considering a MCDM problem that has n criteria and m possible decisions and each criteria has an associated weight w which signifies its perceived relevance. With F being a

$m \times n$ matrix of the possible decisions and their values for the given criteria, the score S for decisions is given as:

$$S_i = \sum_{j=1}^n w_j F_{ij} \text{ for } i = 1, 2, \dots, m \quad (2.13)$$

This model is a strong choice in a single dimensional problems (Aruldoss et al. [2013]), and it's simplicity makes it very easy to understand and implement. However, it is difficult to adapt to multidimensional problems.

This model was used by Aznaoui et al. [2021] in the proposed geographic adaptative fidelity routing protocol for gathering data in wireless sensor networks. An optimized weighted sum model is used to solve this problem, where the maximum remaining energy and minimum distance are considered as essential criteria to elect an active leader for each virtual grid to reduce the energy dissipated.

2.6 Conclusions

Mobile networks, like many other domains of engineering, are very complex systems that are in constant evolution. Some of the most recent steps in that evolution are NFV and SDN, both new ways to decouple network logic from the hardware and make modern networks much more flexible. And this flexibility creates unprecedented opportunities to implement self-healing systems.

Of course, one major trade off of that flexibility is the increase in failure points and failure varieties. 2021 alone witnessed various different outages in services that rely on some type of network, caused by a myriad of different reasons. And whereas NFV technologies are mostly standardized, the same cannot quite be said about SDN, which poses a non-trivial challenge to any technology that is developed on top of them to be widely adopted, even if there are platforms such as OpenFlow that are far more popular than their alternatives.

These technologies are fundamental to a different breed of mobile network operator: the MVNO. Without their own RAN, MVNO are inherently bound to developing their products and technology on top of NFV and SDN deployments, which is no less the case for Truphone.

The increasing volume of traffic and network complexity means that outages will only get more frequent and often harder to fix³¹, meaning that costs and/or downtime are bound to only go up for expert-reliant approaches, and this calls for a new automated paradigm. This paradigm is self-healing.

Self-healing is divided in three major components: detection, diagnosis and compensation/recovery. Detection identifies outages, diagnosis understands the outage and compensation/recovery attempts to return the network to

³¹2022 Internet and network outages will continue to get worse before they get better: <https://www.emarketer.com/content/2022-predictions-internet-network-outages-will-continue-worse-before-they-better>, Accessed 20/01/2022

a healthy state. There is also a great potential in leveraging data mining and machine learning techniques to create more efficient mechanisms of self-healing, as shown in literature.

Chapter 3

Related work

This chapter aims to give a review of existing work and common-practices regarding outage evaluations and self-healing components such as outage detection, outage diagnosis and outage compensation/recovery

It is divided into two major sections: Outages in connectivity services and Self-healing Systems. The former explores methodology developed for outage impact evaluation, as well as some RCA approaches. The latter explores the three components of Self-healing previously defined (detection, diagnosis and compensation/recovery). This section gives noticeably more emphasis to Outage Detection because it is not only the most researched of the three components but also the part of the system that can best be shared with other different systems, whereas, for example, compensation/recovery is a rather specific step.

Finally, the chapter concludes with a brief reflection of the analyzed works and what conclusions can be drawn that will bolster this project.

3.1 Outage Analysis Methods

This section explores work done for outage analysis, namely in the form of impact evaluation and root cause analysis. These methods are important to assess the relevance and source of an outage, which itself is important to determine its priority when it comes to chose which outage type the self-healing system will focus on resolving.

3.1.1 Outage Impact Evaluation

Aceto et al. [2018] consider that outage impact evaluation can be non-formal, user-centric or network-centric. Table 3.1 presents a summary of the perspectives presented in Aceto et al. [2018].

Perspective of Evaluation	Advantages	Difficulties	Existing Literature
Non-formal	-	Do little to contribute to the accurate understanding of the impact of network outages	Cho et al. [2011], Dainotti et al. [2011], Internet Without Borders, Brookings Institution
User-centric	Simple to understand and deploy. Also, they seek global Internet reachability	Do not consider partial outages, which account for most disruptions	Eriksson et al. [2013], Horrigan [2010], Djatmiko et al. [2013]
Network-centric	The use of formal metrics makes comparison between studies a possibility	No widespread adoption of a defined set of metrics	Li and Brooks [2011], Wu et al. [2007], Liu et al. [2013], Agarwal et al. [2010]
Network- and user-centric	A more complete perspective of an outage event, with the advantages of both network- and user-centric perspectives	No widespread adoption of a defined set of metrics	Dainotti et al. [2012], Benson et al. [2013]

Table 3.1: Perspectives and their advantages and disadvantages

Non-formal Impact Evaluation

Cho et al. [2011] analyzed reports from the NTT group about the 2011 Tohoku earthquake that detailed how many base stations, transmission lines and circuits for fixed line services were damaged and how was voice call acceptance affected in Japan. The Internet's reachability was measured through volumes of traffic recorded by ISPs in the area of the quake, a measurement that was recognized by the authors as flawed because it could not factor in power outages, as well as server shutdowns and other infrastructural disables, scheduled for maintenance and repair. Other non-formal evaluations of outage impacts include the analysis done by Dainotti et al. [2011] of the Arab Spring regarding the effects of censorship in the affected countries, as well as analyses done by Non-Government Organizations (NGO) such as Internet Without Borders and the Brookings Institution regarding human rights violations (discrimination, free speech, abuse of power,...) or even on the economic impacts of Internet outages in the prosperity of nations.

User-centric Impact Evaluation

In this method, impact is measured from the perspective of the end-users of a network. For example, Eriksson et al. [2013] evaluate outage impact considering the estimated population that is served by the concerning networks, as per the correlation between population density and Internet usage established by Horrigan [2010].

Djatkiko et al. [2013] evaluate outage impacts by the number of BGP prefixes that are unreachable, as that translates to a rough number of Internet users that cannot be reached. These approaches are interesting because they keep their focus on global Internet reachability, which is ultimately the objective of outage mitigation. However, this simplistic approach to impact evaluation is not sensitive to partial outages, since it can only detect users affected by full outages where services are not available at all, which is a big downside since partial outages make the majority of Internet accessibility issues.

Network-centric Impact Evaluation

Lastly, network-centric evaluations most commonly attempt to model network behavior before, during and after periods of instability. This can be seen in works such as Li and Brooks [2011] that use network metrics to model healthy behavior and affected behavior. Wu et al. [2007] define two groups of evaluation metrics: Reachability Impact Metrics (RIM), that consider how many paths are unavailable, and Traffic Impact Metrics (TIM), that consider how much traffic is being rerouted due to failing links (and thus causing network congestion).

Liu et al. [2013] propose a centrality metric of an Autonomous System (AS) based on the concept of betweenness of network nodes, which is defined by out of all the existing paths in the network, how many of them use a given node as a routing point. This allows an estimation of the traffic load that goes through a given AS and how important that node is to the network as a whole. This is deemed more effective to estimate a node's load and importance than mere connectivity since connectivity is local and betweenness is global. Similarly, Agarwal et al. [2010] propose a graph-based evaluation method which uses two measurements: number of link failures and terminal reliability (the impact of a failure on a network's connectivity).

Network and User-centric Impact Evaluation

The ideal would be to combine network and user perspectives to gain a more complete picture of an outage's impact rather than a one-sided evaluation. Dainotti et al. [2012] measure the impact of outages by how many IP addresses in the affected geographical area likely lost connectivity. This is done through passive monitoring of data plane unsolicited traffic coming from the affected region. This is, however, a relative measurement because it relies on how many addresses were visible before the perceived outage. This is explored further by using this measurement to define the maximum radius of the impact, which adopts both a user and network centric perspective. Similar IBR-derived metrics were adopted by Benson et al. [2013] to further understand macroscopic connectivity disruptions.

Considerations

Aceto et al. [2018] consider, after analysing the aforementioned approaches, that the vast majority of studies only perform qualitative (rather than quantitative) or non-formal evaluations of outage impacts. These diverging approaches and mostly non-complementary analyses do little to contribute to an exhaustive and accurate understanding of possible consequences of network disruptive events, making studies where user and network centric metrics are used extremely valuable. They call for a framework of common methodologies for not only analysis, but detection, impact quantification, network robustness quantification, risk assessment, and mitigation and survivability techniques. They also mention an increasing interest in the development of subjective metrics, such as Quality of

Service (QoS) and Quality of Experience (QoE), to which they give the latter preference. These metrics can be used to evaluate how an outage affects that which matters most to service providers: the end-user. Furthermore, they mention that whereas small outages in a given network are relatively easy to detect, larger outages involving a myriad of different entities have not yet been properly addressed (and the existing approaches have been found lacking).

Aceto et al. [2018] also state that using knowledge about the underlying network topology is not necessarily applicable to real Internet infrastructure, since gathering knowledge about what is normally a frequently changing topology is not trivial, exemplified by Claffy et al. [2009] and Donnet and Friedman [2007]. They consider that networks containing wireless paths (which we consider to be an inherently critical part of mobile virtual networks) have not yet been studied extensively, at least not as wired wide-area-networks have been, which is evident by the abundance of studies that analyze outages in voice services and use blocked calls as a primary measurement.

3.1.2 Root Cause Analysis (RCA)

RCA is the process of analyzing and identifying existing problems in order to find their root causes. The alternative to this process is to simply put out fires and treat symptoms without resolving the underlying issues of these symptoms. Naturally, RCA is a big part of outage handling in software, network or any other engineering services. This subsection focuses on analyzing automated RCA methods. Table 3.2 shows the works here presented grouped by the type of metrics they target (QoS or QoE).

Target Metrics	Method	Existing Literature
Quality of Service	Decision Tree + Linear Regression	Jain et al. [2016]
	Random Forest or Regression Tree	Pasquini and Stadler [2017]
Quality of Experience	Mean Opinion Score	Letaifa [2017]
	Variable Importance Analysis	Gonzalez et al. [2017]

Table 3.2: RCA works and their target metrics

Jain et al. [2016] propose a two-phase mechanism for QoS prediction in SDN. This method works by correlating Key Performance Indicators (KPI) to QoS metrics through a Decision Tree and then applying a Linear Regression for RCA using these QoS metrics. Pasquini and Stadler [2017] explore this analysis of QoS metrics for application-aware estimation. They propose the use of Random Forest or Regression Trees to estimate QoS based on Video on Demand (VoD) metrics such as frame rate and response time. The estimation accuracy was of over 90%.

Alternative to QoS, QoE metrics have gained much traction as the preferred set of metrics to evaluate the overall health of the service, even though these metrics tend to be more subjective. One of the proposed metrics is the Mean Opinion Score (MOS) which is divided in 5 levels, ranging from 5 (excellent) to 1 (bad) experience levels. Letaifa [2017] propose a method to correlate this QoE

metric to QoS metrics such as loss rate, delay, jitter and throughput, and adjust video streaming parameters to improve QoE.

Gonzalez et al. [2017] propose a method of RCA using Machine Learning and summarization techniques. This approach first collects data surrounding the event of an outage, namely in the form of device alarms. Then, several models are trained to predict whether or not a given outage will happen in a given device (one model per outage). After this, a variable importance analysis is performed as to understand which alarms (and in particular in which devices) were most relevant for the occurrence of an outage. This allows to establish a causality chain between alarms in different devices and better understand why each type of outage occurs.

3.2 Self-healing Systems

Even though Self-healing has been a focus of research for over two decades now, the advent of SDN and VNF make it much more applicable to modern networks due to the programmable nature of these technologies (especially in comparison to the classic hardware-centric paradigm of mobile networks). This field has been gaining traction among the network operator community ever since due to the prospect of replacing expensive and relatively slow manual labor with fast automated programs that can handle large amounts of data at a much larger rate.

Asghar et al. [2018] wrote an extensive and comprehensive survey about self-healing solutions for mobile cellular networks. The works in this survey, however, are mostly for MNO settings where the RAN can be accessed and tweaked. Typically, self-healing mechanisms are divided in three or four major components: Outage Detection, Outage Diagnosis, Outage Compensation and Outage Recovery. Whereas the first three components are always present, the latter is not necessarily present in many of the published solutions. Due to this fact, we aggregate Compensation and Recovery in the same chapter. Asghar et al. [2018] also divide each in whether they handle full outages or partial outages, since the necessary steps to identify one or another are seldom the same.

3.2.1 Outage Detection

In this section, notable work regarding outage detection is presented. The section is divided in works for mobile cellular networks, the Internet, and other relevant settings (such as IoT and SDN scenarios). Table 3.3 summarises the works presented in this section.

Cell-specific Outage Detection

Outage Detection aims to identify (and perhaps even predict) and signal outages through constant monitorization of the network. Asghar et al. [2018] divide exist-

Domain	Methodology	Outage Type	Existing Literature
Cell-Specific	Heuristic	Full	Amirijoo et al. [2009], Liao et al. [2012]
		Partial	Karatepe and Zeydan [2014], Shafiq et al. [2016]
	Supervised ML	Full	Gurbani et al. [2017]
		Partial	Ciocarlie et al. [2014a]
	Unsupervised ML	Full	Zhang et al. [2019], Yu et al. [2018], Zhang et al. [2017]
		Partial	Ma et al. [2013], Rezaei et al. [2016], Ciocarlie et al. [2014b], [Miao et al., 2015]
Internet-specific	Passive	Profile-based	Li and Brooks [2011]
		Time-based	Teoh et al. [2006], Deshpande et al. [2009]
		DP core-based	Schatzmann et al. [2011], Dainotti et al. [2011], Glatz and Dimitropoulos [2012]
		DP edge-based	Tierney et al. [2009]
	Active	Ping-based	Quan et al. [2013], Schulman and Spring [2011]
		Tomography-based	Duffield [2006]
	Hybrid	Full	Katz-Bassett et al. [2008], Xiang et al. [2011], Javed et al. [2013]
	IoT and/or SDN	Blockchain-based	Full and partial
Graph-based		Full and partial	Brandón et al. [2020]
Log-based predictive maintenance		Full	Calabrese et al. [2020]

Table 3.3: Summary of outage detection work presented

ing work into heuristic, supervised and unsupervised approaches. They postulate that machine learning techniques for cell outage detection suffer from errors due to noise in the recorded datasets, and are prone to false negatives in denser scenarios. Not only this, but most approaches (including non-machine learning approaches) need a secondary analysis from a human expert to confirm if an outage does in fact exist, which once more poses an issue in scenarios where outages are a frequent occurrence.

Heuristic approaches deeply rely on pre-existing knowledge of domain experts which makes them very adequate for existing mobile cellular networks. Normally, these approaches apply rule-based systems for outage detection, such as the ones proposed by Amirijoo et al. [2009] and Liao et al. [2012] for full outage detection. The first is a set of thresholds for certain performance metrics whereas the latter is a weighted cost function of performance metrics where a cell is deemed failing if neighbouring cells do not meet the necessary sum threshold for being healthy. These two solutions, however, only detect full outages.

For partial outage detection, Karatepe and Zeydan [2014] proposed an algorithm for cell misconfiguration based on a heuristic algorithm that is reportedly capable of detecting misconfigured cells over 82% of the time. Similarly, Shafiq et al. [2016] proposed a solution that compares cell profiles during routine network operation to periods with a high volume of traffic.

Supervised machine learning-based algorithms are another popular type of approach to outage detection. A great deal of their popularity stems from the facts that these algorithms use datasets curated by experts. For full outage detection, Gurbani et al. [2017] propose a log-based anomaly detection system that uses performance data from normal periods and previously identified outage periods to model anomalous behavior. As such, the data for this model is a set of timeseries. They propose two types of modeling: non-parametric, relying on Chi-

Squared Testing, and parametric, relying on Gaussian Mixture Models (GMM).

For partial outages, Ciocarlie et al. [2014a] propose using a time-series analysis of cell profiles. This is done by modeling a cell's behavior using the AutoRegressive Integrated Moving Average (ARIMA) algorithm and the cell's KPI, and continuously predicting cell states. Whenever a cell's state differs enough from the predicted values (that in theory correspond to the cell's normal state values), this behavior is classified as an outage. The authors also propose using different techniques such as empirical cumulative distribution functions and SVM with radial basis function kernels. The evaluation results suggest that whereas this method can accurately detect partial outages, it always took five hours or more to detect the outage.

An emerging approach to outage detection and arguably the one to garner most attention in recent years is through unsupervised learning methods. Notably, Zhang et al. [2019] propose the use of GAN for synthetic data generation to offset the usually imbalanced nature of outage datasets. Adaboost is used to validate the results of the resulting calibrated dataset. Yu et al. [2018] and Zhang et al. [2017] use network statistics to train a Local Outlier Factor (LOF) classifier, which, in theory, is semi-supervised algorithm that attributes each object a degree of being an outlier factor, as stated by Breunig et al. [2000].

Chernov et al. [2015] do an extensive comparison of clustering algorithms to detect sleeping cells, a prominent problem in Long-Term Evolution (LTE) networks. A cell is considered sleeping when it is in an outage state but does not produce any alarms. The compared methods were KNN, Self Organizing Maps (SOM), Local-Sensitive Hashing and Probabilistic Anomaly Detection. For evaluation, the Receiver Operating Characteristic (ROC) and precision-recall curves have been used, where the Probabilistic Anomaly Detection had the best scoring according to both metrics. Local Sensitivity Hashing was the fastest model to train (linear order) and Probabilistic Anomaly Detection was the fastest to detect sleeping cells. Ma et al. [2013] propose using Dynamic Affinity Propagation for the same problem, resorting to the Silhouette index as a quality criterion for clustering. This approach was clearly successful in detecting sleeping cells in the test data, but Asghar et al. [2018] consider that its performance in real world data may be influenced by users suffering deep fade.

Even though SOM is perhaps one of the most popular unsupervised clustering techniques used, Rezaei et al. [2016] compare the usage of chi-squared automatic interaction detection, quick unbiased efficient statistical tree, Bayesian networks, SVM and classification and regression trees, finding that the SVM has the best detection rate among the supervised learning techniques, even though it is the one that takes the longest to train. Quick unbiased efficient statistical tree, however, has the shortest training time with an also very high detection rate. Ciocarlie et al. [2014b] use topic modeling to detect outages in a cellular network, where each cluster is assigned a outage or non-outage meaning according to domain knowledge, and [Miao et al., 2015] use a kernel-based LOF anomaly detection approach and suggest that this method can better deal with non-uniform distributions compared to the regular LOF method.

Internet-specific outage detection

Aceto et al. [2018] provide an interesting dissection of Internet outage detection techniques by dividing existing work in passive and active detection as such:

- **Passive detection**

Passive detection are usually based on control plane information by analyzing data collected via the BGP protocol. Some alternative methods explore data plane traffic by analyzing volume variations and correlating them to events. However, it is widely recognized by the authors of these approaches that it is not trivial to guarantee user privacy, and that this remains an open issue, with some authors attempting to establish general approaches that resort to Multi-Party Computing (MPC) (Djatkiko et al. [2013]). Some approaches are:

- **Profile-based:**

I-seismograph, a tool developed by Li and Brooks [2011], detects network outages and evaluates their impact by modelling the normal state of the Internet and then monitoring the network for a given window to measure if and how the Internet's behavior changed from normalcy. This model leverages public BGP data to measure and define what are normal attributes and compare fixed time windows to check for anomalous behaviors

- **Time-based change detection:**

Teoh et al. [2006] created a tool for RCA of BGP anomalies that works by clustering BGP updates related to the same events and attempts to analyze the root cause of these anomalous events by considering an Internet-centric perspective (by analyzing how these events affect relationships between different AS) and a home-centric perspective (by analyzing how an AS is affected by an event originating in another AS several hops away). Deshpande et al. [2009] created a mechanism that detects and analyzes routing instabilities that may or may not be caused by network outages;

- **Data plane core-based:**

Schatzmann et al. [2011] proposed FACT (Flow-based Approach for Connectivity Tracking) that relies on flow-level data exported by border routers of a network to compare incoming and outgoing traffic flows. The system collects NetFlows records and aggregates flows per remote host, network or AS. This way, the increase in unsuccessful outgoing one-way connections and the decrease in two-way connections can be used to detect outages. Aside from this, Dainotti et al. [2011] and Glatz and Dimitropoulos [2012] proposed methods that leverage unsolicited data plane traffic to detect outages.

- **Data plane edge-based;**

An entirely different approach by the name of PerfSonar was proposed by Tierney et al. [2009] which is a collaborative network monitoring

platform providing several network troubleshooting tools deployed in several independent networks to share data like SNMP counters, useful for detecting outages.

- **Active detection:** Active detection (or probing) is also a popular method of collecting data for outage detection. Ping and *traceroute* are by far the most popular tools for this purpose, periodically scouring the network from several different vantage points. Tomography is used to a lesser extent and is based on leveraging prior knowledge of the measured networks and its topology, but this can be difficult because often the measured network is the whole Internet).¹.

It is important to note that while these solutions are also effective, most existing work suffers from poor scalability due to the fact that active probing translates to a non-trivial amount of communication overhead (using the existing bandwidth to transmit packets necessary for these solutions). There are several active detection approaches, namely:

- **Ping-based;**

Quan et al. [2013] proposed Trinocular, a system that probes each IP block with ICMP echo requests (pings) at 11 min intervals and classifies responses as positive (in the event of the reception of an ICMP reply) or negative (in the absence of a response or in a response indicating the address is unreachable)². This system's principles had been explored by Quan et al. [2012] and Quan et al. [2014] by using this approach to investigate macro-events such as the hurricane Sandy, the Sanriku earthquake and the Egyptian revolution (all of these events happened in 2012). Another interesting approach was proposed by Schulman and Spring [2011] by the name of ThunderPing, a tool that leverages pinging and *traceroute* analysis to measure the connectivity of Internet hosts before, during, and after forecasted periods of severe weather.

- **Tomography-based;**

Duffield [2006] proposed a system that relies on coordinated end-to-end probes to analyze if specific links between network elements have been broken. This system requires devices to collaborate by providing ICMP responses, and is especially helpful when networks are configured to discard ICMP messages and in general when the overall layout of the network is known.

- **Hybrid passive-active detection:**

Perhaps the most promising method of Internet outage detection, hybrid approaches attempt to combine the advantages of both passive and active outage detection while minimizing the downsides of each. By primarily adopting passive measurements for information acquisition and resourcing only to opportunistic active probing, the scalability issue of active detection

¹These projects often rely on global measurement infrastructures like Archipelago (<https://www.caida.org/projects/ark/>, Accessed 20/01/2022)

²It is important to notice, however, that Trinocular has the peculiarity of increasing by 0.7% the amount of Internet unsolicited traffic for the networks where it operates

is reduced to an acceptable standard. Some works done in this scope are the following:

– **BGP, *traceroute* and ping hybrid;**

Katz-Bassett et al. [2008] proposed Hubble, a system capable of detecting Internet reachability problems, namely when packets do not reach the destination network through the data plane even though routes for that network exist in the control plane (according to the BGP data). As mentioned previously, unreachability was defined as a situation where a prefix is unreachable or very slow to respond from several different access points. Xiang et al. [2011] proposed Argus, an equally hybrid system that is used to detect IP prefix hijacking. Javed et al. [2013] proposed PoiRoot, a hybrid system that works in real-time to allow any ISP to accurately isolate the root causes of path changes in their prefix routing. It uses BGP data combined with *traceroute* analysis for this effect.

IoT and/or SDN approaches

Khan et al. [2019] propose a blockchain-based approach for detection and prevention of DDoS attacks on SDN controllers for smart grids. This approach relies on keeping a decentralized list of all malicious traffic sources and another for possible victimized components through blockchain. Identifying the attackers and the victims is done heuristically, through a threshold of maximum allowed traffic.

Brandón et al. [2020] propose a graph-based approach to RCA with the aim of identifying outage affected regions. This method generates graphs of the system, where nodes are network elements and edges are network connections. Then, a subgraph search is made to identify regions that compare to previously identified anomalous regions. These previously anomalous regions are annotated by the authors as regions that were suffering outages.

Another relevant field of research for this project is log-based predictive maintenance, which usually applies to industrial settings with IoT capabilities. Calabrese et al. [2020] proposed a log-consuming event-based machine learning architecture for predictive maintenance in an IoT scenario for a woodworking corporation. As per the abstract, "predicted failures probabilities are calculated through tree-based classification models (Gradient Boosting, Random Forest and Extreme Gradient Boosting) and calculated as the temporal evolution of event data."

Liu et al. [2015] proposed a method that leverages Random Forests to decide what detection approach should be used and with which parameters, using data labelled periodically by network operators. The goal of this approach is to aggregate the many existing detection approaches and reduce the time operators must spend labelling data.

3.2.2 Outage Diagnosis

Whenever an outage is detected, it is necessary to identify what type of outage it is (for example, is it hardware or software failure? Was it caused by human interaction, a cyberattack or by something else?). This step may seem trivial, but modern networks can have such a wide plethora of outage causes that linking very specific scenarios to their causes is unfeasible. Instead, researchers have strived to create systems capable of automatically assigning outage characteristics to specific diagnoses, an approach better suited to the dynamic nature of outages in networks. Table 3.4 contains a summary of all the work presented in this section.

Method	Outage Type	Existing Literature
Heuristic	Full	Szilágyi and Nováczki [2012]
	Partial	Shafiq et al. [2016]
Supervised	Full	Khanafer et al. [2008], [Chen et al., 2019], Ciocarlie et al. [2014b]
Unsupervised	Full	Rezaei et al. [2016]

Table 3.4: Summary of outage diagnosis work presented

Szilágyi and Nováczki [2012] propose utilizing expert knowledge to create targets for network KPI to diagnose full outages. The model uses the difference between the real and target values in a weighted sum to calculate a diagnostic score, which is then associated to a range of different scores, each attributed to a different cause. The proposed technique was validated in real world data and was able to diagnose each outage correctly. Shafiq et al. [2016] present a method for partial outage diagnosis by building network profiles before, during and after network events. The analysis done by the authors depends solely on expert interpretations of the data. They postulate that faulty behavior is more prone to happening when users connect to the network in a uncoordinated fashion, which normally is not a problem because networks are usually designed with this in mind. But with major events and gatherings, additional capacity may be necessary for the network to endure this stress.

The relative simplicity of heuristic solutions allows them to be easily interpretable and reverse engineered by experts, as well as often less computationally intensive than more advanced approaches. These factors make them very solid choices in exploratory work. However, they can be prone to becoming obsolete due to the shifting dynamics of live networks ever greater in dimension and traffic, according to Asghar et al. [2018]. Instead, learning base solutions are much more flexible and can adapt to these changing dynamics. Khanafer et al. [2008] propose to identify possible hardware causes for full outages given their "symptoms" (failures) using a Naive Bayes Classifier and discretizing the KPI values in ranges, and it was naturally found that entropy minimization binning was naturally more effective than percentile discretization.

[Chen et al., 2019] propose AirAlert, a system designed to detect outages and identify their root cause in a cloud service system. It works as a global

watcher for the entire cloud system, collecting all alerting signals and proactively predicting outages that may happen. In this system, diagnosis is done by leveraging a Bayesian network to correlate alerting signals with outage types by conditional dependence and outages prediction is done by using a gradient boosting tree based method.

Ciocarlie et al. [2014b] propose the use of Markov Logic Networks (MLN) and Principal Component Analysis (PCA) to diagnose network outages related to weather and misconfiguration issues. This approach, however, heavily depends on expert knowledge to initialize the MLN weights.

Rezaei et al. [2016] propose an unsupervised clustering approach to fault diagnosis and compare it to other approaches such as expectation minimization, density-based spatial clustering of applications with noise, agglomerative hierarchical clustering, X-means and k-means clustering. The results of this unsupervised approach are then validated by experts and by Silhouette Coefficients, where they show the best results out of all the other approaches.

Asghar et al. [2018] state that outage diagnosis is an under-explored aspect of self-healing in mobile cellular networks, at least when compared with detection and compensation techniques, especially for partial outage diagnoses. They attribute this to the fact that similar partial outages can originate from different root causes, and suggest that future work should focus on exploring existing outage databases without creating artificial outages, since real world applications of supervised learning solutions require a compilation of databases of real outage scenarios to be effective.

3.2.3 Outage Compensation

Once an outage has been diagnosed, the next step in the mechanism is to take action and leverage network resources to minimize the impact failing components may have on the rest of the network and on service quality. Alternatively, instead of compensating, the mechanism may try to take direct action to fix the failing component, for example in the event of incorrect parameterization. Table 3.5 contains a summary of all the work presented in this section.

Context	Existing Literature
Software Defined Networks	Kuźniar et al. [2013], Zhou et al. [2017], Qiu et al. [2019]
Sequential Outages	Mirkhanzadeh et al. [2018]
Software Aging	Paing [2020]
IP Network Recovery	Harada et al. [2014]

Table 3.5: Summary of outage compensation/recovery work presented

In the case of SDN, compensation/recovery actions can be very context specific. For example, Paing [2020] propose a mechanism that uses rejuvenation controllers and Stochastic Reward Nets to predict outages related to software ag-

ing, which happens when software applications must run uninterrupted for long periods of time and end up consuming so many resources that problems like memory leaks start to happen. The solution to software aging is through rejuvenation, which can be full or partial. This method predicts when applications are likely to suffer aging problems and when is the appropriate time to apply rejuvenation, as well as, if full rejuvenation is needed, estimating the downed time of that component.

Harada et al. [2014] propose Another Improved Multiple Routing Configuration (AIMRC), a method to reroute a failure component for fast IP network recovery and compare it to more conventional methods named Multiple Routing Configuration (MRC) and Improved Multiple Routing Configuration (IMRC) (Imahama et al. [2013]). According to the authors, the AIMRC method, when compared to MRC and IMRC methods, it decreased the average total number of increased hops about 40% and 10%, respectively. However, the AIMRC method also increased the total number of entries in routing tables about 15% and 25%, respectively. They propose as future work to investigate a method to deal with multiple component faults.

Mirkhanzadeh et al. [2018] propose PRONet, a two-layer (Ethernet and Wavelength Division Multiplexing (WDM)) Research and Education Network with mechanisms capable of protection (implemented on the Ethernet layer) and restoration (at the WDM layer). This mechanism is prepared to handle and overcome a first outage and prepare the network for a second outage. This was implemented in a manufacturing application.

Kuźniar et al. [2013] propose an approach that, on the event of an outage, creates a new SDN controller instance that runs in an emulated environment consisting of the network topology without the failing network elements. The system then replays the last inputs observed by the failing elements to generate a network state that accounts for the failing elements and installs the difference rule set between the real world state and the emulated state.

Zhou et al. [2017] propose Delorean, an approach that acts as a shim layer between SDN applications and the SDN controller. In the event of a bug or application crash, Delorean traces the cause of this event (this is referred to as the crash path) by iterating through the history of input events to determine the best rollback point for the application's state. When this happens, a transformation manager produces several different code paths and ranks them in order of optimality, which is seen as a compromise between quickness of the rollback and efficacy of the rollback. Finally, the system iterates through these different code paths until it finds one that does not lead to a crash.

Qiu et al. [2019] propose an algorithm that exploits pruned searching to quickly compute recovery paths for all-pair switches/hosts upon a link failure in an SDN context. They also extend this algorithm to quickly find the shortest guaranteed cost path for applications requiring stringent path robustness levels, which ensures that the recovery path used upon on-path link failures has the minimum cost. Compared with traditional solutions, their evaluations show that the algorithm is approximately 8 times faster than the practical implementation

and approximately 1.93 times faster than the state-of-the-art solution.

3.3 Conclusions

Evaluating the impact that an outage has on the overall network's performance is an important task and one that does not as of yet possess a standardized approach or framework that allows different outages to be comparable.

Self-healing, as it exists in the domain of mobile networks, still has much to be explored in the context of SDN, especially when it comes to diagnostics and compensation strategies. Most solutions are very context specific and do not necessarily translate well to different outages in different scenarios because the problems they solve are very different in nature and context.

However, much work has been published when it comes to RCA and outage detection in the SDN context. In these tasks, while machine-learning based or assisted approaches are achieving positive results, heuristic approaches are still very relevant and prove to be the best solution to many problems. Data collection and labelling can prove to be a difficult and complex task, with no clear way to avoid some operator input, which further difficults the use and proper evaluation of machine-learning approaches.

Chapter 4

Understanding Outages at Truphone

This chapter describes the study of outages at Truphone, which was core to the Business Understanding and the Data Understanding components of the CRISP-DM methodology employed, aforementioned in subsection 2.4.1. The ultimate goal of this study was to gain a better understanding of Truphone's outage history and how could a self-healing system solve existing issues and generate value, mainly through saving man hours that are currently being spent solving outages.

The research done in order to understand Truphone's outage history and methodology is detailed in following sections. This includes the comparison with categories already applied by Truphone engineers with the categories found during the state of the art analysis. Two main avenues of research are detailed: expert knowledge extracted primarily from interviews and incident room archives, presented in Sections 4.1 and 4.2 respectively, and a comprehensive analysis of Truphone outage records (via JIRA tickets), presented in Section 4.3.

This study was an iterative process: Expert input helped refine the analysis of JIRA records, and the analysis generated talking points and questions for the interviews. Over the course of the six-month scholarship, there were a total of 36 interviews/meetings dedicated to discussing this project. There were also several punctual and informal conversations to address specific issues, as well as the recurring meetings of the R&D team that often included discussion's about the project's status and work to be done. It is important to note that the analyses presented in Sections 4.1, 4.2 and 4.3 were performed concurrently and the order of their presentation is not relevant because they all reference each other.

The study resulted in understanding how outages in Truphone as an MVNO differ from what was studied in the literature, which were primarily cases in MNOs. Furthermore, a rationale for selecting outages that are good candidates for self-healing was derived from this analysis. This rationale would then be applied in the selection of an outage to develop a prototype self-healing mechanism for. These results are presented in subsection 4.4.

4.1 Expert knowledge

Inevitably the one of the most influential factors in any stage of this project is expert knowledge and input, manifested regularly in the daily meetings with the R&D team and punctually in meetings with specific Truphone members that play key roles either in this project or the processes of outage handling at Truphone, either from a technical standpoint or from an impact standpoint.

Table B.1 of Appendix B shows some of the key elements in the development of the thesis and how they contributed to the process of analyzing and categorizing outages.

This section describes some of the most important meetings held, their objectives, participants and outcomes. It also explains any insights obtained from these meetings.

4.1.1 Outage stakeholders' focus group

One important example of this initiative to involve and interview experts taken a focus group-type meeting with many of the potential stakeholders in the outage handling/mitigation process, which included but were not limited to engineers that are called to resolve/mitigate outages when they happen, product owners and team managers. This interview is highlighted because it grouped stakeholders of various positions, responsibilities and perspectives of the outage handling process.

There were a total of twelve attendees. In the meeting, members with the following roles had some type of input:

- One Principal Engineer
- One Front Office Lead
- Two Front Office Engineers
- One Truphone Web Services Engineer
- One Network Service Manager
- One R&D Software Engineer
- One Technical Support/Service Quality Manager

There were several objectives in this meeting:

- Introduce the project to members of Truphone that were considered to benefit directly or indirectly from a self-healing solution and/or recognized as key elements in Truphone's current outage handling methodology;

- Validate the analysis done so far of the JIRA dataset, and obtain suggestions as to how that analysis could be fine tuned;
- Obtain the experts' perspective of where a self-healing system would be most impactful and, conversely, which types of outage could be best-suited for a self-healing system;
- Present examples of outages that had been selected as possible candidates for self-healing during the preliminary analysis (JIRA and Incident Rooms) to understand their validity in the current Truphone context;
- Clarify a number of questions that had come up during the preliminary analysis.

In order to prepare this meeting, a slideshow was to assist in addressing the defined objectives, and a number of questions designed not only to clarify existing doubts but also to stimulate discussion between the participants were formulated:

- What outages do you think are best candidates for automatic detection and recovery?
- Are things like "runaway processes" and "capacity exhaustion" good candidates for automatic detection and recovery?
- Are all third party outages unfixable/unpredictable?
- How do you usually decide the priority of an outage (Low, Med, Hi, Crit)?
- How relevant are old outages?
- Some tickets have upwards of one year from when they are created to when they are resolved. Why?
- When an outage appears, can you make an estimation of how long it will last? Or is it usually hard to predict their duration from their onset?
- Why are alarms "muted" sometimes for very long periods of time?
- Is there any automatic recovery system in place in Truphone?
- What are the types of reconfiguration that can be automated?
- Since Root Cause Details are usually extensive texts, can you suggest terms that are common in relevant outages?

A result of this meeting was the suggestion that the original JIRA dataset, on which the initial analysis of Truphone's outages was performed, was prone to include inaccuracies or in vague information in tickets due to the filters used in its collection. It was suggested that Salesforce data may prove more consistent and detailed to perform a global analysis of outage data. It was established that there was a considerable technological upgrade in Truphone's network infrastructure

in late 2020/early 2021, and as such many of the issues that were relatively common prior to that date were not relevant enough today. Furthermore, the mechanism for outage impact calculation is detailed in Salesforce, which means that filtering tickets using Salesforce information would provide a more consistent and robust dataset with much less noise, with the caveat that this better-filtered dataset is inevitably less comprehensive of Truphone's outage landscape than the original one.


The subject of network observability and alarms was also broached during this meeting. In Truphone's infrastructure, there are some alarms and traps that directly correlate to outage-like events, but these are not being handled automatically, relying instead in human action to be tagged and raised as an issue. The process of uniforming the work that is now usually done by humans into machine-interpretable meta-information such as flags in order to enable the automation of manual processes was also something that was planned, but was still in a preliminary stage of development. While there are no systems in place that analyze alarms and tag them as outage-like events or causes of such events, there are people doing that work manually due to a non-uniform method of categorizing alarms. It was also suggested that a closer-look to the manual process of analyzing outages could be beneficial to this project, and log-related alarms were hinted as a potentially good place to start looking for a potential outage to fix.

There were also some problems that were singled out as potentially interesting targets for compensation/recovery, such as horizontal pod auto-scaling in Kubernetes environments and Cron jobs that begin misbehaving. Runaway processes were considered an issue of the past that might not be relevant anymore and capacity exhaustion issues were considered potentially interesting problems with possibly very hard solutions.

The discussion provided valuable improvements and a better understanding on the collection and analysis of JIRA's data, and even though it did not lead to one conclusive candidate of an outage, it excluded some types that were in consideration such as runaway processes, confirmed the interest in some types such as capacity exhaustion and gave way to further meetings with members of the FrontOffice team which is largely responsible with monitoring and handling outages. Furthermore, all participants confirmed that a self-healing system was very relevant in various aspects of Truphone's technology, albeit with varying degrees of applicability and difficulty. However, most pre-planned questions were not approached because the discussion took a more organic form where participants discussed between themselves rather than directly reply to the moderator.

4.1.2 Meeting about Salesforce and how it impacts JIRA data

This meeting was had with a Network Service Manager who was responsible with dealing with network outages, among other things. The objective of this meeting was to gain a better insight into the JIRA ticket dataset (see Section 4.3), namely regarding what distinguished common JIRA tickets with tickets with an associated Salesforce report, which was argued to augment both the quality and



		Impact			
		Priority	Low	Medium	High
Urgency	High	Medium	High	High	
	Medium	Low	Medium	High	
	Low	Low	Low	Medium	

Basic Impact, Urgency & Priority matrix

Figure 4.1: Impact, Urgency & Priority Matrix Example (Source: bmc.com, Accessed 15/8/2022)

the content of a ticket, and thus make the report of that outage more reliable and insightful.

In the event of an outage, this person and their team had to identify the outage, evaluate impact and urgency and devise a mitigation action. After the outage was mitigated, they would move on to discovering the outage’s root cause and resolve it as to return the affected elements to their original, pre-outage state, if possible. It is important to stress that outage mitigation is priority when the outage occurs, and only after mitigation actions were taken does the team look at long-term fixes for the issue. Aside from this, they were also responsible with communicating with the relevant service and product teams, and possibly with affected clients as well, depending on the outage’s priority and external communication policies. Another part of this responsibility was documenting and reporting every process associated with the outage handling both in Salesforce and JIRA.

This team had a well-defined methodology for handling outage reports. Impact, one of the defining attributes of an outage, was defined by a priority matrix where the priority level is assigned from P1 (highest impact) to P4 (lowest impact) as a function of both impact (how serious are the effects of the outage) and urgency (how quickly do the clients need to access the affected services). It was suggested that there should be several priority matrices, for different teams, outages and services. An example of a priority matrix can be found in Figure 4.1¹.

Furthermore, the "Service" and "Subcategory" fields are filled with values out of a predefined list of statistically relevant categories that allow operators to understand exactly what was the perceived problem – unlike the general JIRA ticket, where there is no predefined list of values and they are filled at the operator’s full discretion.

¹Source: <https://www.bmc.com/blogs/impact-urgency-priority/>, Accessed 15/8/2022

4.1.3 Meetings with the FrontOffice team

The FrontOffice team is the team tasked with monitoring the network and its alarms and dealing with eventual issues and outages. These responsibilities make them not only clear benefactors and stakeholders from any eventual self-healing mechanism, but also very empirically knowledgeable about Truphone's outage history and record.

With this in mind, the FrontOffice team was regularly consulted whenever some clarification about existing issues was necessary, and there were several meetings held with members of this team. The meetings of 29/3/2022 and 30/3/2022 set important groundwork for the discovery of the outage that would be targeted by the self-healing system.

The 29/3 meeting had the objective of discussing a promising outage candidate for self-healing. It was held with the FrontOffice Lead and a FrontOffice Engineer. This outage consisted of a periodic configuration file transfer from machines in Amsterdam or London to machines in Australia sporadically resulting in the corruption of the transferred files, causing an important service to crash in these machines. This outage was a good candidate because the issue was well understood by experts, there was already a mitigation strategy which as capable of automation and there was abundant data for development and testing. Posterior to this meeting, this outage was communicated to the Principal Engineer which, after analyzing the affected components, determined that the root cause category of the outages was misconfiguration. After this intervention and as a direct result of the research, the outage was resolved by the principal engineer.

The 30/3 meeting had two principal objectives: understand FrontOffice's approach to network alarms and to find another promising candidate for self-healing. It was held with the FrontOffice Lead and a FrontOffice Engineer. Both objectives were accomplished. The outage to be analyzed was an issue with a technology called Temporary Service Access Number (TSAN), that enables calls from Truphone numbers to be redirected through an intermediate number, rather than passing straight from caller to receiver. This technology is useful for products such as TMR, among others, that due to technological constraints must make use of an intermediate number to function. The identified outage was caused by the pools of intermediate numbers being sporadically unreachable, disabling any calls that need to use this technology. This unreachability was caused by a number of reasons, all of which outside the ownership of Truphone. The problem was detected via hourly reports of TSAN availability, where FrontOffice engineers defined a threshold for healthy and unhealthy behavior of the pools. The usual mitigation was switching the affected TSAN pool to one that was not affected and was regionally close (see Section 4.4.3).

4.2 Incident Rooms

The standard procedure for dealing with outages at Truphone is to open an incident room in a Microsoft Teams dedicated channel that will be used by the personnel tasked with analyzing and resolving the outage to discuss ongoing events and analyses, as well as sharing and recording important pieces of information during the process. Much of the information that is shared during these incident rooms is often used to fill out the JIRA tickets that correspond to the same outage that is being analyzed. The usual workflow for an incident room is presented in Figure C.1 of Appendix C.

4.3 JIRA dataset

JIRA² is a proprietary issue tracking platform developed by Atlassian that allows for agile project management and has bug tracking features. This platform is used by Truphone members to keep track of their tasks during sprints as well as to keep track of incidents and their resolution. Naturally, when an outage occurs, JIRA tickets are created to describe the issue by detailing how it was detected, how does it impact the business and who “owns” the issue. An example JIRA ticket sourced from Herbold et al. [2020] is presented in Figure 4.2. After its creation, the ticket then gets filled with information about the steps taken to resolve the situation, namely fields containing what was the root cause and how it was fixed, as well as a thread of comments that engineers post containing important information they might have found during their analysis. JIRA also supports exporting tickets to different file formats such as .xlsx and .csv, where the user can select which fields to export and filtering criteria for which tickets should be included (such as time periods, specific field values, etc.).

4.3.1 Generating the dataset

The ticket export feature was used to generate a dataset of 4,041 tickets that reported potential outages or network issues since the beginning of 2019. Later, this dataset was reduced to 947 outages collected since the beginning of 2021 due to advice from experts, that hinted towards the fact that there were major network changes in the beginning of 2021 so the outage landscape would be different prior and after this date, rendering many outages that happened before this not relevant anymore. Some of the fields included in this dataset are Ticket ID, Duration, Priority, Summary, Level of Impact, Reason for Case, Problem Root Cause, Fix, Service and Subcategory, among others. These fields were particularly relevant because they allow for a general overview of Truphone’s outages’ characteristics, although it should be noted that there are some reliability issues with the data provided. Sometimes the duration of a ticket may not correspond to the outage’s

²Atlassian’s webpage for JIRA: <https://www.atlassian.com/software/jira>, Accessed 30/8/2022

Figure 4.2: Example of a JIRA ticket (Herbold et al. [2020])

true duration, fields such as the priority and level of impact are heavily influenced by the bias of the person that fills the ticket out and sometimes that person may not be technically knowledgeable enough to distinguish between different problem root causes, often opting to introduce vague values such as “None”, “Other” or “Unknown”, which are usually reserved for unknown issue causes or outages that have not been seen before.

Furthermore, some JIRA tickets had Salesforce³) data associated (see Section 4.1.2). This data included information such as the assigned priority according to a priority matrix, an incident grade, how the issue was detected and the number of affected users. As explained in Section 4.1.1, there was added value in analyzing this subset of the original JIRA dataset, referred to as the Salesforce dataset, to obtain better insights about the outage paradigm at Truphone. It is important to note, however, that Salesforce platform was never used, only JIRA.

4.3.2 Preprocessing the dataset

Since the collected dataset contained many fields that were manually filled and there was useful information that could be extrapolated from existing fields, it was necessary to work the dataset into a more palatable form.

Some of the preprocessing work necessary to make the collected dataset more usable was:

³SalesForce is a Customer Relationship Management (CRM) platform, designed to facilitate customer-business interaction and relationships. Although it was not used directly in this dissertation, it indirectly influenced the research. Salesforce’s webpage: <https://www.salesforce.com/eu/products/what-is-salesforce/>, Accessed 30/8/2020

- Removing punctuation and capitalization from text fields allowed to account for possible misinputs/typos from tickets, as well as granting an extra layer of uniformity in the dataset. For example, it was not uncommon in fields like "Problem Root Cause" to have some tickets with the value "other" and others with "Other".
- The field "Service and Subcategory" was separated into "Service" and "Subcategory" to granularize the dataset's information for posterior analyses.
- Fields that possessed dates, such as the "Outage Start Time" and "Outage End Time" fields, were parsed into epoch timestamps to allow the generation of a new field "Outage Duration".
- Certain fields such as "Problem Root Cause" could be filled with values such as "other" or "unknown", which although valuable to understand both how many root causes were deemed inconsistent with existing categories and how many outages were not fully understood at the root cause, were not very informative when it comes to analyzing existing categories employed by Truphone experts. It is important to note that when an outage occurs, there is always an effort to prevent it from happening again. Thus, tickets that possessed these values were marked as "vague" to facilitate posterior analyses.
- Since there were some tickets with two fields that represented the assigned priority (one for the priority assigned in JIRA and another for Salesforce and they used different notations (JIRA uses the values "Low", "Medium", "High" and "Critical", while Salesforce uses "P4" to "P1"), they were uniformized to JIRA's notation.
- A list of common typos was compiled, such as "manuall" or "work around" rather than "workaround", and forcibly corrected on categorical fields such as "Root Cause" or "Fix".

Aside from this, there was also an attempt at automatically extracting outage categories from existing information using methods such as TF-IDF vectorization and LDA topic modelling.

The TF-IDF method was applied to obtain a ranking of *ngrams* (sets of n words) according to their relevance in the field of "Root Cause Details", which is usually a field of long form text explaining the cause of an outage. It was configured using Python's `sklearn`⁴ package to search for *ngrams* with $n \in [2, 4]$ with a minimum of 5 appearances in the corpus and presence in a maximum of 60% of the corpus. These parameters were chosen after some experimentation as they were the ones that presented a more diverse and informative group of *ngrams*. The reason why single words were not considered was because more often than not the terms presented by TF-IDF were non-informative of category, such as names of machines, mentioned experts or service providers, so they were discarded all together in lieu of the adopted approach. The top 100 ranked *ngrams*

⁴scikit-learn: Machine Learning in Python: <https://scikit-learn.org/>

<i>ngram</i>	rank	<i>ngram</i>	rank
"alarms triggered"	3	"jira closed"	49
"capacity exhaustion"	6	"kpn imsi"	50
"cf close"	9	"lack space"	53
"clear enough/intuitive"	11	"let know"	54
"closed sp"	12	"loss connectivity"	57
"closing sp"	13	"memory exhaustion"	63
"configuration error"	15	"memory leak"	64
"connection error"	17	"new version"	68
"connectivity failure"	18	"planned activity"	72
"cx req"	20	"planned maintenance"	73
"data sessions"	22	"product catalogue"	74
"data traffic"	23	"rate plan"	77
"default value"	24	"running space"	80
"details comments"	25	"service impact"	82
"disk space"	28	"service recovered"	83
"experience clear"	30	"service restored"	84
"experience clear enough/intuitive"	31	"signalling stack"	87
"ha proxy"	33	"sip invite"	90
"hello team"	35	"sw bug"	94
"internal action"	41	"user data"	98
"ip change"	43	"user experience"	99
"issue happened"	45	"user experience clear"	100
"issue solved"	47		

Table 4.1: TF-IDF most significant *ngrams*

after removing every *entry* that contained the word "cause" (due to them usually not being informative) are presented in Table 4.1.

Even though some useful *ngrams* such as "capacity exhaustion", "connection error", "configuration error" or "planned maintenance" can be informative of the problem's root cause, there are still many *ngrams* that are uninformative of the root cause, such as "alarms triggered", "cf close", "clear enough/intuitive". This approach needs greater fine tuning to allow for an automatic extraction of outage root causes.

The LDA approach was not as successful in extracting potential root causes. The model was configured using Python's `gensim`⁵ to pass 50 times over the corpus of the field "Problem Root Cause". A total of 20 topics were generated.

The results can be found in Table 4.2. It is important to note that the terms were stemmed to use their base forms. As per Chai [2022], the low word probabilities for most topics suggest that there is a certain difficulty in finding differentiating terms. This type of topic modelling would require extensive fine-tuning to make viable in this use case.

Ultimately, after reflecting on the preliminary results of this automated

⁵gensim: Topic modelling for humans: <https://radimrehurek.com/gensim/>

	probability*word				
0	0.021*"connectivity"	0.015*"power"	0.015*"failure"	0.014*"vm"	0.012*"lost"
1	0.026*"sim"	0.021*"change"	0.019*"table"	0.017*"ims"	0.016*"profile"
2	0.067*"vodafone"	0.042*"date"	0.021*"team"	0.020*"caused"	0.014*"bics"
3	0.053*"account"	0.020*"com"	0.018*"wrongly"	0.016*"service"	0.016*"usag"
4	0.068*"script"	0.025*"memori"	0.021*"vpn"	0.020*"sims"	0.020*"threshold"
5	0.038*"user"	0.027*"plan"	0.025*"space"	0.023*"high"	0.022*"data"
6	0.020*"mss"	0.020*"bics"	0.019*"traffic"	0.018*"pivotel"	0.017*"ip"
7	0.080*"error"	0.078*"bug"	0.036*"database"	0.034*"field"	0.026*"order"
8	0.023*"kafka"	0.019*"enable"	0.017*"cod"	0.015*"suspected"	0.013*"waiting"
9	0.041*"success"	0.027*"executed"	0.019*"resource"	0.016*"hw"	0.013*"pool"
10	0.029*"service"	0.020*"caused"	0.017*"impact"	0.016*"truphone"	0.014*"traffic"
11	0.106*"fix"	0.097*"code"	0.041*"details"	0.015*"duplicate"	0.014*"value"
12	0.040*"issue"	0.029*"capacity"	0.026*"traffic"	0.023*"exhaust"	0.016*"app"
13	0.033*"errors"	0.015*"provided"	0.014*"input"	0.013*"sip"	0.013*"cause"
14	0.086*"issu"	0.063*"n"	0.052*"test"	0.041*"jira"	0.021*"sp"
15	0.026*"post"	0.017*"bics"	0.014*"manually"	0.014*"ldn"	0.014*"links"
16	0.103*"cause"	0.085*"root"	0.029*"internal"	0.027*"related"	0.025*"failur"
17	0.059*"configuration"	0.048*"default"	0.029*"access"	0.027*"profil"	0.026*"issues"
18	0.046*"cerillion"	0.038*"db"	0.028*"problem"	0.022*"restart"	0.019*"caused"
19	0.115*"missing"	0.023*"hss"	0.019*"present"	0.017*"db"	0.016*"misconfiguration"

Table 4.2: LDA first 20 identified topics and their respective 5 most probable terms

approach, the necessary effort estimated for its further exploration and development was deemed too great for the benefits it would ultimately bring, and thus this avenue of research did not prove successful and was abandoned in lieu of a more manual study of existing categories.

Additional analyses included querying fields relative to outage root causes for potentially troublesome machines via regex, since machine names follow a preset convention, as well as aggregating tickets by "reporter" (the person who created the ticket) and "assignee" (the person responsible for handling the outage). The results of this query are not presented here due to their sensitive nature, but they were later used for expert interviews, namely the latter to identify major stakeholders in the outage handling process.

4.3.3 Truphone’s outages and state of the art categories

One of the first steps to understand Truphone’s outage data is to compare it to the state of the art categories. Truphone already has an internal method of characterizing, detailing and reporting the event of an outage through tickets, including things like problem root cause and fixes, but it can sometimes be inconsistent between outages and variable in terms of what information is presented. For example, tickets may or may not have an impact described, and this description can be a ticket field categorizing the impact between "none", "intermittent loss" or "full loss" or it can also just be part of the general description and only analyzable manually.

Some of the more relevant categories for outages’ root causes used in Truphone reports are the following:

- Configuration error – A service, network element or device was improperly configured and thus rendered unusable, inoperative or malfunctioning.
- Capacity exhaustion – The allotted resources for a service or function were exhausted. Some examples of this are bandwidth for connections, memory leaks resulting in excessive memory usage, excessive CPU usage (for example, in runaway processes), among others.
- Software failure - A bug, crash or any other type of software malfunction that affects services and elements negatively that renders them unusable, inoperative or malfunctioning.
- Outage (third party) – A sudden loss of connections, resources or services provided by third parties. For example, a power outage in a third party data center that is being used by Truphone Web Services can lead to an outage of Truphone services.
- Connectivity – Issues that affect customers’ capability of communicating with other users. This can affect voice, text or data.
- Dirty data – Data that was improperly inserted, corrupted or otherwise unfit for consumption by the systems that require them that might affect users’ capacities of using Truphone’s services.
- HW failure – A hardware malfunction that renders services and functions unusable, inoperative or malfunctioning,
- Customer error/User error – Improper usage of Truphone’s services by customers that result in customer support engagement and thus tickets being created.
- Change (internal) – Updates, upgrades or other types of internal changes to existing services, functions or systems that can result in unexpected anomalous behavior, rendering them unusable, inoperative or malfunctioning.
- Change (third party) – Updates, upgrades or other types of changes to existing services, functions or systems provided by third parties that can result in unexpected anomalous behavior, rendering them unusable, inoperative or malfunctioning by Truphone.
- Spam fraud – Reception or dispatch of an excessive number of communication attempts (calls or text) in a short time span. The sender is usually at fault.
- Provisioning error/failure – Error in provisioning SIM profiles to devices (usually related to the eSIM technology).
- Design error – Improperly designed services or functions that result in customer having difficulty or incapability of operating them. Usually applied to customer facing systems.
- Unsupported feature – Customers’ attempts or requests to use a feature that is not yet supported by the services or functions they are using.

- Congestion – Excessive network traffic that consumes the allotted bandwidth for a given region.
- System failure – Similar to HW failure.
- Power failure – A hardware failure that was caused by power-related issues, such as power outages, malfunctioning power supplies, cut power cables, among others.
- Incorrect access privileges – Attempt to use services or functions for which the user does not have permission.
- License/certification expired – Certificates that have expired.
- Poor coverage – Truphone’s network partners have poor coverage in a certain region or area, resulting in Truphone’s services or functions having low connection quality.
- Other/Unknown – None of the causes above. This is used either when it is a very specific problem that cannot be aptly described by any of the above causes or the operator that filled out the ticket did not know or was not able to assign the proper cause. This is also the most prevalent value for the field "Problem Root Cause".

Some of these categories may intersect with each other i.e., "Power Supply" may be a subset of "Outage (third party)", which is due to there being several levels of granularity within these categories. This allows for more common subsets of certain categories to be uniquely identified, whereas less common subsets are assigned the less granular, overarching category.

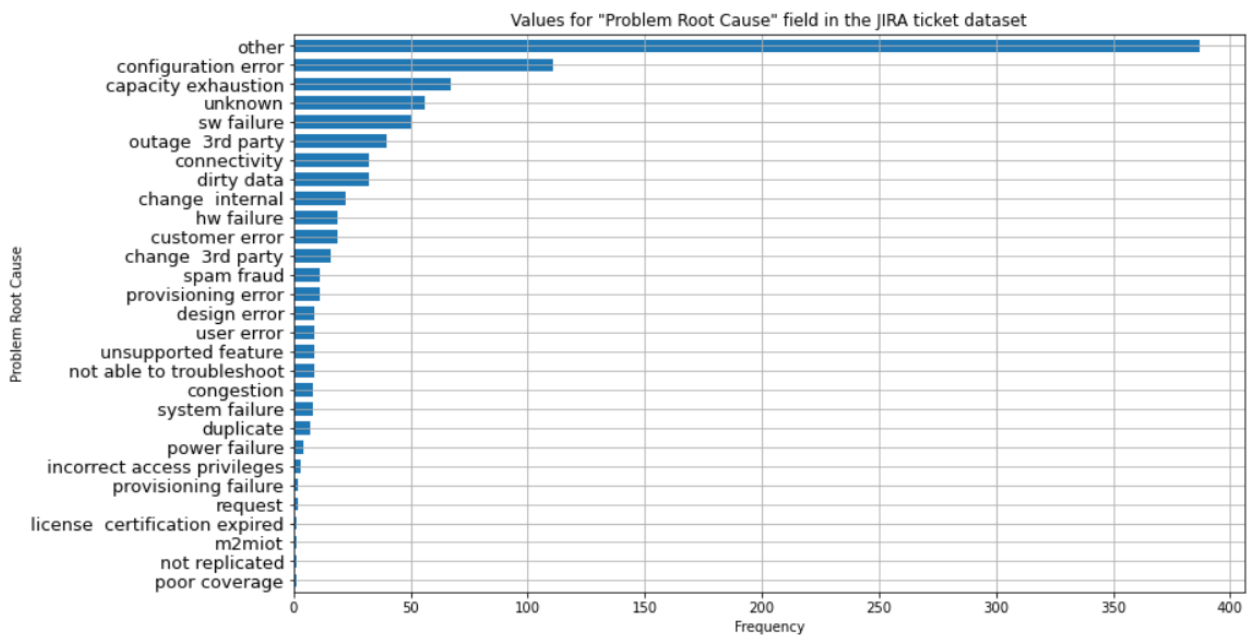


Figure 4.3: Values for "Problem Root Cause" field in the JIRA ticket dataset

The distribution of 947 outage tickets collected since the beginning of 2021 is presented in Figure 4.3, demonstrating that there is a large amount of tickets

that are vaguely categorized, with classes such as "other" or "unknown" representing a large quantity of tickets. A connection between the categories used by Truphone staff to report outages with the state of the art categories can be found in Table 4.3.

SoA category	Reference	Truphone category
Network Failures	Donegan [2016],Gunawi et al. [2016], SRPSDVE	Capacity Exhaustion, Connectivity
Physical Link Failures	Donegan [2016]	Connectivity
Network Congestion	Donegan [2016],Li et al. [2013], Gunawi et al. [2016],SRPSDVE	Congestion
Customer Device Issues	Donegan [2016]	Customer Error/User Error, Incorrect Access Privileges
Misconfigurations	Donegan [2016],Li et al. [2013], Gunawi et al. [2016]	Configuration Error
Server Issues	Donegan [2016], Li et al. [2013], Gunawi et al. [2016], SRPSDVE	Configuration Error, SW Failure, HW Failure
Malicious Damage	Donegan [2016], SRPSDVE	Spam fraud
Network/Service Enablers	Donegan [2016], Gunawi et al. [2016]	Outage (3rd party), Change (3rd party)
Cybersecurity Issues	Donegan [2016], Gunawi et al. [2016], SRPSDVE, Steinberger et al. [2015]	Incorrect Access Privileges
Power Outages	Li et al. [2013], Gunawi et al. [2016], SRPSDVE, [Trang and Hong, 2021]	Power Failure
Network Hardware Issues	Li et al. [2013], Gunawi et al. [2016], SRPSDVE, Asghar et al. [2018]	HW Failure, System Failure
Software Issues/Bug	Li et al. [2013], Gunawi et al. [2016], SRPSDVE	SW Failure, Design Error, Unsupported Feature, Incorrect Access Privileges
Data Issues	Li et al. [2013], Gunawi et al. [2016], SRPSDVE, Jesmeen et al. [2018]	Dirty Data, Incorrect Access Privileges
Memory Leak	Li et al. [2013], [Avritzer et al., 2020]	Capacity Exhaustion
Upgrades/Updates	Gunawi et al. [2016], SRPSDVE	Change (Internal)
Third Party Issues	Li et al. [2013],Gunawi et al. [2016], SRPSDVE	Outage (3rd party), Change (3rd party)
Human Error	Li et al. [2013], Gunawi et al. [2016], SRPSDVE	
Other Types		Spam Fraud, Provisioning Error, Other, License/Certification Expired, Unknown

Table 4.3: Connection between state of the art categories and categories reported in Truphone

It is important to notice that some times Truphone categories do not possess a 1:1 relationship with state of the art categories. This is because often the true category of an outage can only be understood by analyzing the context of the outage and its nature, so some categories overlap simply because of the vagueness of assessing categories without their proper context. Nevertheless, a clear correlation between categories employed at Truphone and those extracted from the state of the art research is present. Also, the analyzed data did not present any cybersecurity-related outage, and thus there is no Truphone category for cybersecurity outages. This does not mean, however, that Truphone has not suffered any cybersecurity-related incidents, but rather that possibly none of these incidents have resulted in network/service failures or degradations. Furthermore, the Human Error category is considered very ambiguous since the majority of the aforementioned categories can be caused by erroneous human input, so it was not assigned any corresponding Truphone categories. Finally, the "Other Types" category was reserved for outages that were either not present in the state of the art, such as the "Provisioning Error", or categories such as "Other" or "Unknown". It is important to distinguish between these last two categories because

"Unknown" means the problem could not accurately be diagnosed, but "Other" usually means that the problem was not possible to be summarily described using pre-established categories. This is somewhat subject to operator input, and thus some "Other"-labeled outages could be assigned a more proper category, but an in-depth analysis of "Other"-labeled tickets revealed that, in general, these are very specific problems which are hard to attribute a pre-existing category.

A manual categorization of tickets collected in 2022 was performed in order to have an understanding of outages in Truphone that more closely correlates with the literature. For that purpose, the physical vs. logical divide in Aceto et al. [2018] was combined with the root cause categories represented in Table 4.4.

Categories	Description	Source
Power Issue	Equipment failure due to power outage/power supply issues	Li et al. [2013], Gunawi et al. [2016] Trang and Hong [2021], SRPSDVE
Hardware Issue	Equipment failure for reasons not related to power supply	Donegan [2016], Li et al. [2013], Gunawi et al. [2016], Asghar et al. [2018], SRPSDVE
Security Issue	Physical or cyberattacks that intend to compromise services	Donegan [2016], Steinberger et al. [2015] Li et al. [2013], Gunawi et al. [2016], SRPSDVE
Storage Issue	Issues with data storage systems or databases	Gunawi et al. [2016], Li et al. [2013]
Dirty Data	Incorrectly formatted or inserted data that compromises the normal functioning of systems/services	Li et al. [2013], Jesmeen et al. [2018]
Update	Update to existing products or services that break or disable them unintentionally	Li et al. [2013], Gunawi et al. [2016]
Congestion	Excessive general traffic in the network that causes services to stop functioning at satisfactory quality	Donegan [2016], Li et al. [2013], Gunawi et al. [2016], SRPSDVE
Misconfiguration	Systems that are improperly configured accidentally	Li et al. [2013], Gunawi et al. [2016]
Planned Maintenance	Necessary maintenance procedures that disable services and systems	SRPSDVE
Bug	Unexpected behavior in software that disables systems/services	Li et al. [2013], Gunawi et al. [2016]
Capacity Exhaustion	Databases, links or computing resources that are exhausted by the systems that require them	Truphone data
Memory Leak	Unexpected excessive use of memory by software processes	Li et al. [2013], Avritzer et al. [2020]
Design Issue	Features of software products rendered unusable by customers due to design flaws	Truphone data
Connectivity Issue	The connection either between the client and the network or between network elements is not working properly or at all	Li et al. [2013], Gunawi et al. [2016], Donegan [2016], SRPSDVE

Table 4.4: Outage source categories in Truphone data

This categorization was applied to 100 different tickets in 2022, and the identified root causes can be found in Figure 4.4. Furthermore, the distribution of the priorities that were attributed to these tickets can be found in Figure 4.5.

It is important to notice that the percentages presented were rounded down to the closest integer and only categories with a higher than 1% presence were displayed in Figure 4.4. Both of these decisions were made for the sake of visualization.

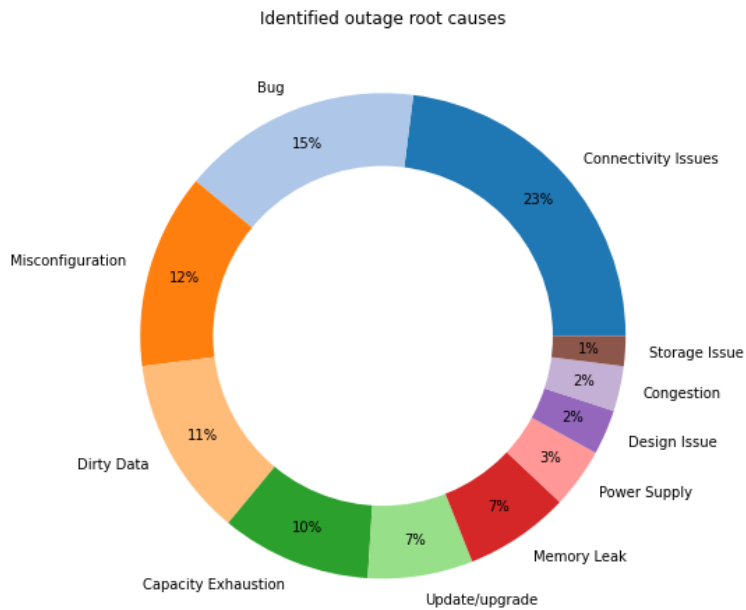


Figure 4.4: Identified root causes

One key take away of these priorities is that since JIRA's priority default is "Medium" by design, so that it is only changed when there is a relevant reason to do so. However, it makes this analysis heavily biased by tickets that did not have their priority changed at any point in their existence. To gain a less biased perspective of these priorities, an analysis restricted to tickets with Salesforce data attached was made. These tickets are filled out in a much more thorough fashion due to reporting events of higher relevance or severity, but not all outages or network problems are given this thorough analysis, and some sectors of Truphone's infrastructure do not get this type of analysis at all. Nevertheless, the fact that these tickets are guaranteed to have an thoroughly comprehensive analysis by Truphone experts grants them much value. The distribution of the priorities attributed in Salesforce for tickets in this dataset can be found in Figure 4.6.

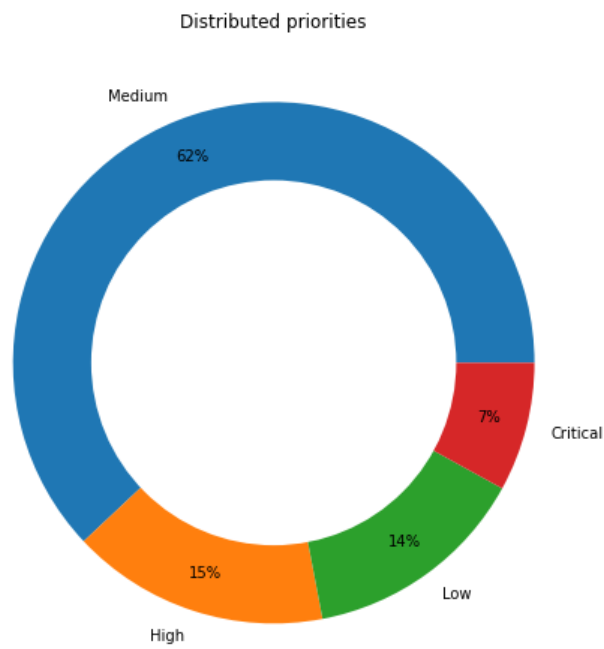


Figure 4.5: Assigned priorities

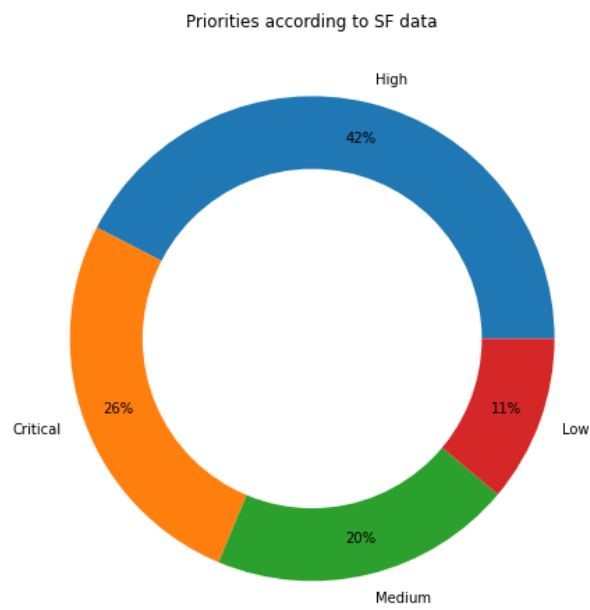


Figure 4.6: Priorities as attributed in the Salesforce dataset

4.4 Results

The research of Truphone's outage history and methodology shed some light into not only which problems existed in Truphone and how they were solved but also which possessed characteristics that favored an automation of their solution. Additionally, the consideration of an automatic way to process JIRA tickets and generate insights should be kept in mind, as it showed interesting results albeit crude, and given more time would have been a promising avenue of research.

During the outage type selection phase, and before selecting the outage that would eventually become the de facto target of the self-healing prototype, there was another outage that was briefly analyzed as a candidate (described in section 4.4.2). However, it was quickly fixed by Truphone's principal engineer after an initial report in a daily meeting, as it turned out to be a misconfiguration issue (that nevertheless consumed man-hours of the FrontOffice Team on a weekly basis). So as a direct result of simply researching and internally communicating specific outages, a problem was fixed, man-hours were relocated to more productive efforts and thus Truphone now does not spend money to mitigate this recurring issue.

4.4.1 Viable outage types

Based on the research, interviews and analysis performed, there are several constraints when determining whether an outage is fit for self-healing, namely:

- outages that are too rare are not well suited to self-healing because an automated handling system would most likely not have any meaningful impact in terms of resources saved. Therefore, outages should occur with some regularity
- it is also important that the outage has some degree of impact in the business to maximize the value added by developing an automated self-healing solution
- the outage should have abundant data available, ideally labelled (or with an available labelling scheme) to facilitate testing any produced solution
- its mitigation plan is well understood and defined by experts, and is capable of automation

Additionally, priority is given to outages that are considered urgent and impactful. Both of these factors can be provided by a priority matrix as presented in Figure 4.1, which Truphone uses only for specific products.

Even with an automatic outage categorization mechanism, identifying whether or not an outage fulfills these criteria must be done manually, because within the same categories, outages can vary widely in terms of regularity, impact and data abundance. A mitigation plan may also not be available, implementable

or automatable by every outage within a category, as some outages pose a significant challenge in mitigation by the MVNO. A good example of this was an outage that occurred in January 2022 when there was a power outage for one of Truphone's Data Centers. This power outage had massive repercussions in several Truphone products and services. The site experienced black and brown outs due to external power grid instability in voltage and frequency which not only disrupted equipment but also triggered unexpected behaviours in the local redundancy infrastructure. This affected all site systems, from Truphone's compute to physical site access management.

Even with existing redundancy in Truphone's global architecture, an outage of this caliber still heavily disrupts Truphone's normal functioning and its solution is far from predefined due to how complex and far-reaching the outage was.

Furthermore, one of the main criteria for the evaluation of the worthiness of an outage for self-healing was how many man hours were spent fixing them, which naturally gives priority to more frequent outages.

4.4.2 First outage candidate

The first outage candidate was caused by a recurrent corruption of configuration files that had to be sent to a core network element in Australia on an hourly basis. This corruption, in turn, made the configuration files unusable and disabled a service called FreeRadius, which disabled Data and Voice over LTE (VoLTE) for the following hour (until a new configuration was sent) or until it was manually fixed by engineers.

The problem was well understood because the configuration files followed a convention, and when they were corrupted the rules of that convention were broken. This made identifying corrupted files (and where they were corrupted) easy and returning them to their uncorrupted state possible. There was also an abundance of collected uncorrupted files and a few corrupted examples, and it would be possible to collect more corrupted files in the future, since this problem occurred on average on a bi-weekly basis.

This brief preliminary understanding was relayed to the R&D team and the principal engineer quickly took action to fix the issue, eliminating the outage. The problem was that the transfer mechanism used sometimes corrupted large files. This was not detected when the system was originally built but with business growth it recently started happening more and more frequently.

The solution to the problem was activating a compression method in the transfer mechanism to compress the transferred files up to ten times, solving the current issue. Hence, as a direct result of the candidates' research, a problem that regularly compromised Truphone services and consumed engineers' man-hours was solved, and a new candidate for self-healing was researched.

4.4.3 Description of the selected type

The problem that ended up being selected for self-healing was an outage in TSAN pools. These outages occur roughly between once every two weeks to once a month, and generally result in clients in a specific geographic region being unable to use services that depend on TSANs, such as TMR which is a flagship product of Truphone, until the issue is fixed manually by a FrontOffice engineer.

What is TSAN

TSAN is a technology used by Truphone to enable certain products and services, prime among which is TMR. It is important to understand this technology because it was the service targeted by the developed self-healing prototype.

This technology, not owned by Truphone but provided as a service by several providers, allows, in gross terms, the redirection of a call between two cell numbers through a third number. This can be seen as a call from an A number to a B number being redirected through a C number. This C number is a temporary number that is not assigned to any specific client, but rather belongs to a pool of numbers that are used solely for this purpose. It is a critical component of TMR and when it fails, some TMR calls also fail.

Of important note is the pre-call flow of a call that makes use of a TSAN. A normal TSAN call begins with a CAMEL Application Part (CAP) Initial Detection Point (IDP), which contains information about a request for a call, such as the caller address (which is the A number, and the person trying to make a call), the called address (which is the B number, and the person receiving a call) and the Forward-to-Number (FTN) (which is the C number, and the TSAN). For the call to be successful, there must be a corresponding Session Initiation Protocol (SIP) Invite no more than 12 seconds after the IDP. In this SIP Invite, there is only a caller address (which is still the A number) and a called address (which must be the C number for this invite). If the SIP Invite arrived on schedule, it is then followed by another CAP IDP (known as a "retrigger"), where the caller address is the A number, the called address is the C number and the "ftn" is the B number. Only if these three signals arrive on schedule is the call successful, otherwise it is considered a failed call. This flow is illustrated in Figure 4.7.

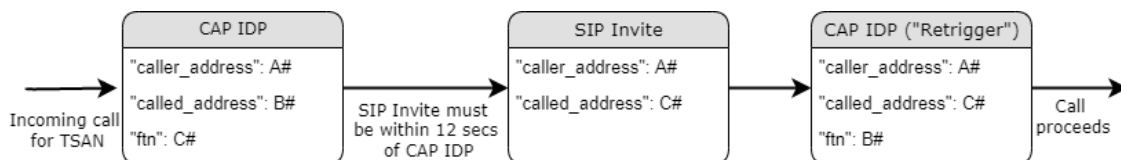


Figure 4.7: TSAN pre-call flow

Rationale for selected type

As aforementioned, this outage temporarily disables the use of products such as TMR for a region of Truphone customers. This effect could be heavily disruptive

to businesses that, for example, need to record phone calls made, like financial services such as banks, insurance companies, et cetera. This disruption, in turn, degrades the perceived quality of the services provided by Truphone, which, as with any disruption of service, strains company-client relations and costs both Truphone and its costumers valuable time and money. In addition to this, a group of Truphone engineers needs to take time out of their usual functions to handle and solve the outage. This is not only costly for Truphone in man-hours directly spent solving and "babysitting" the issue, but also in hours that these engineers could have spent in other tasks. The TSAN-related outages present in the JIRA dataset that occurred in 2022 are present in Table 4.5. There are three types of issues:

- Outages, where the service is disrupted because there are failures in call attempts;
- Reporting issues, where the existing monitoring mechanism failed to deliver reports about TSAN pools;
- Capacity exceeded, where all numbers of a given TSAN pool were being used, potentially causing disruptions

All of these issues are in the scope of the self-healing mechanism, because it is intended to detect and mitigate outages and potentially report them, even if they are caused by capacity exhaustion, because the system does not look for root causes.

This outage happened 27 times between January and August 2022 and the average reported man-hours spent per outage to solve the issue were 8.23 (7.99) hours. There were, however, many tickets that did not report how many man-hours were spent (valued as "N/A"). It is clear that there is a large variance in how long an outage can take to solve, and that reporting issues are usually the quickest to solve.

Another factor that made this outage very appealing as a candidate for self-healing was the fact that it was generally well understood by experts. Although understanding the root cause of the issue posed a significant challenge because it required understanding why the host MNO was failing to provide service, the symptoms for the issue were well defined and visible for Truphone (calls began failing), which allowed detection, even if forecasting the problem was hard. The mitigation plan was likewise understood and defined, and there was even a perceived potential for automation of this mitigation plan.

Finally, this outage uses signalling data for which there is an Amazon Web Services (AWS) S3 bucket with all Call Detail Records (CDR) generated since the beginning of 2021. This meant that there was an abundance of unlabeled, raw data available to enable further research and development of the problem.

Date	Man-hours spent	Type
03/Aug/22	4	Outage
31/Jul/22	4	Reporting issue
19/Jul/22	N/A	Outage
17/Jul/22	N/A	Capacity exceeded
15/Jul/22	N/A	Outage
14/Jun/22	4.5	Outage
08/Jun/22	25+1	Outage
05/Jun/22	1.5	Reporting issue
28/May/22	N/A	Outage
18/May/22	5	Outage
17/May/22	N/A	Capacity exceeded
12/May/22	N/A	Capacity exceeded
10/May/22	N/A	Capacity exceeded
28/Apr/22	N/A	Capacity exceeded
27/Apr/22	2.75	Outage
20/Apr/22	N/A	Capacity exceeded
18/Apr/22	17.5	Outage
20/Mar/22	N/A	Outage
08/Mar/22	N/A	Capacity exceeded
08/Mar/22	4.5	Reporting issue
02/Mar/22	1	Reporting issue
23/Feb/22	2	Outage
22/Feb/22	N/A	Capacity exceeded
10/Feb/22	N/A	Outage
07/Feb/22	24	Outage
04/Jan/22	10	Outage
03/Jan/22	8.5	Outage
Avg +/- (std): 8.23 +/- (7.99)		

Table 4.5: TSAN-related outages present in the JIRA dataset that occurred in 2022

Current methodology for handling of the selected type

Since this problem already existed since the adoption of the TSAN methodology, Truphone engineers already had a mechanism in place for detecting and mitigating it, albeit not totally automated. As with the proposed self-healing solution, it is heavily reliant on the concepts explained in subsection 4.4.3.

The FrontOffice team has a script that monitors an Astellia⁶ database with CDRs of many signalling protocols, CAP and SIP among them, on an hourly basis. Every hour, this script counts IDPs and "retriggers" and internally calculates

⁶Astellia is a leading provider of network and subscriber intelligence enabling telecom operators to drive service quality, maximize operational efficiency, reduce churn and develop revenues. Its vendor-independent real-time monitoring and troubleshooting solution optimizes networks end-to-end, from radio to core." Quoted from <https://innovacom.com/company/astellia/>, Accessed 18/8/2022

total call attempts and call failures. From this, it then calculates a success rate (SR) as per equation 4.1.

$$SR = \frac{CallAttempts - CallFailures}{CallAttempts} \quad (4.1)$$

If at any given time a pool has a failure rate of 10% or higher, that pool gets flagged as a potential outage. From there, either the FrontOffice team or a network engineer performs tests on the networks using SIGOS⁷ in batches of calls to clarify if there is an outage on a given pool or if it was a false alarm. If the expert does not think the results of these calls were sufficient to draw conclusions from, they can perform more batches of calls until they feel confident enough in a decision regarding the status of the TSAN pool.

If these batches of calls confirm that there is an outage in a pool (over 10% of calls failing), not only do the network engineers contact the provider of the TSANs to clarify and resolve the issue, but they also temporarily switch the affected pool's numbers with numbers from another healthier pool. The decision of what pool should replace the affected pool is based on expert knowledge and intuition, but they report that the main criteria for this decision are the following, roughly in order of priority:

- Numbers available^{*8};
- Geographical proximity;
- Call price*;
- Used call protocol*;
- Call quality*;

The amount of available numbers is an eliminating factor, meaning if a pool fails this criterion it is completely removed from consideration, since the new pool should be able to accommodate all calls destined to the affected pool. Geographical proximity is very impactful in QoE for the customer, because a greater geographical distance means a greater "lag" in any transmitted signal, causing customers to feel a delay in the call. Call price is also understandably a factor that should be minimized. This is, however, a metric that neither network engineers nor the wholesale team know outright. Rather, they need to confirm prices in chosen regions when they make a pool substitution, and often choose standard pools because of this factor.

⁷SIGOS is a platform that allows for "scalable and fully integratable telecommunications testing solutions" (<https://www.yumpu.com/en/document/view/2900628/keynote-sigos-product-guide>, Accessed 8/9/2022).

Formerly owned by Keynote System, SIGOS was acquired by Mobileum in 2022: <https://www.mobileum.com/about/news-press-releases/mobileum-inc-acquires-sigos/>, Accessed 8/9/2022

⁸*in the replacement pool

Used protocol is an operational factor, because not all pools use the same protocols. Some pools use CAP while others use ISDN User Part (ISUP)⁹, which is an older, less desirable protocol because it is less flexible. Examples of regions that still use ISUP rather than CAP are Spain and Hong Kong.

Additionally, call quality is perhaps the hardest factor to assess in a pool, because it involves manual testing after the pool change has been effected to verify that the quality is up to par. In case it is not, the change is rolled back and another pool is chosen as a substitute.

The pool substitution consists of sending a new configuration file to something called a Rhino Element Node (REM)¹⁰. These nodes are responsible for managing which numbers belong to which pool, and they follow a configuration file with a rule set for this. To change pools for a given region, the pool of that region needs to be "redirected" to the numbers of a pool from a different region, so that when calls from the original region require the use of a TSAN, they get redirected to the numbers of the new region.

Ultimately, after the mitigation has taken place, the changes eventually need to be reverted back to the network's normal state. This rollback is done when the affected numbers are confirmed to be working normally again, either by tests done by network engineers or by communication of the number provider. Rolling back the changes made is as simple as uploading another configuration file to the REM node.

Disadvantages and issues of the current methodology

Even though the current methodology employed to handle the TSAN outage is effective in its purpose, it nonetheless possesses a number of disadvantages that could be mitigated or even negated by a functional self-healing mechanism.

First and foremost, it is a process heavily dependent on manual input from experts in everything from testing to mitigation. Naturally, not only does this cost Truphone many man-hours to fix (see Table 4.5), but it also takes some time to fix. Ideally, with an automated system, both man-hours spent and downtime required to address and mitigate the issue would be greatly reduced, increasing the quality of experience for customers such as important financial clients of mobile recording services like banks or insurance companies.

The detection method also has a myriad of associated reliability issues. For example, the mechanism being used only correlates CAP IDPs and retriggers, and does not consider at all SIP Invites. While this approach is not necessarily incorrect because usually when a call fails, there is no retrigger, it may miss a specific subset of call failures. The method also uses an older data source that will soon be replaced.

⁹Introduction to ISUP: https://www.dialogic.com/webhelp/msp1010/10.2.3/webhelp/MSP_DG/ISUP-Introduction_to2.htm, Accessed 20/8/2022

¹⁰REM User Guide: <https://docs.rhino.metaswitch.com/ocdoc/books/rem/1.5.0/rem-home/>, Accessed 21/8/2022

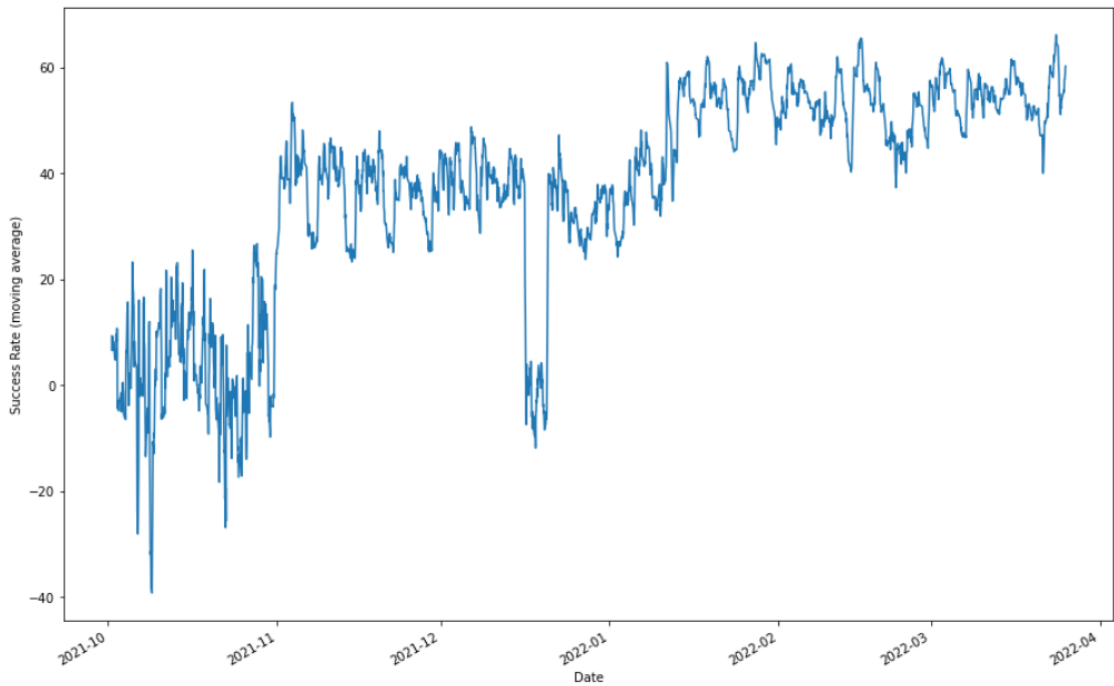


Figure 4.8: Moving average of success rate for TSAN related calls from July 2018 to May 2022 in Spain

There are also edge cases where call and failure counts are miscalculated and result to negative numbers in the success rate which, while easy to detect, require network engineers to verify with test calls. Evidence of this reliability concern can be found in Figure 4.8 which is a visualization of TSAN related calls from October 2021 to March 2022 in Spain, smoothed by using a rolling average with a window of 168. In this period, there was some instability in the detection mechanism because success rates often averaged below from the 90% expected from a healthy pool, even though there was no indication that Spanish pools were suffering a prolonged outage. Part of this can be due to the fact that in periods with little to no calls, the mechanism's default would be to assign a success rate of 0. This results in the calculated success rates averaging at 0.71 ± 0.28 . Several cases similar to this were found during analysis, suggesting it was not an unique event, but it was also not frequent. It is nevertheless crucial to note that the FrontOffice team and network engineers are fully aware of this limitation and have manual processes designed to address it, meaning that this issue causes no disruption on the current methodology, but raises difficulties in automating the process.

Furthermore, the detection mechanism does not look at pools as they are defined in the REM nodes but rather at global titles¹¹ from where the call originated, as well as the first n digits of the TSAN used. This is due to the configuration and operational models being different, which poses a challenge to potential integration of detection and mitigation automations.

¹¹Mobile global titles are unique identifiers for use within Public Land Mobile Networks (PLMNs) to identify the country, PLMN and HLR/HSS: <https://ldapwiki.com/wiki/Mobile%20Global%20Title>, Accessed 22/8/2022

Chapter 5

Self-Healing Framework

This chapter describes the research and development of a prototype self-healing framework for the TSAN outages problem. This outage temporarily disables the use of products such as TMR for a region of Truphone customers which could be heavily disruptive to businesses that, for example, need to record phone calls made, like financial services such as banks, insurance companies, et cetera. This disruption, in turn, degrades the perceived quality of the services provided by Truphone. This process began in early May 2022, and concluded in August 8th, 2022, when the scholarship concluded. This phase contributed to every phase of the CRISP-DM model, with Section 5.1 contributing to Data Understanding and Data Preparation, Sections 5.2, 5.3 and 5.4 contributing to Modelling, Section 5.5 contributing to Evaluation (whose results are presented in Section 5.7) and Section 5.6 contributing to Deployment, albeit with just a high-level plan. All phases indirectly contribute to Business and Data Understanding because, as this problem was further researched, a better understanding of the associated technologies, issues and concepts was obtained.

5.1 Data collection and preprocessing

The first step in the development of this mechanism was obtaining CDR data to analyze. This data was obtained from an AWS S3 bucket containing all signalling data since the beginning of 2021. Only CAP and SIP CDRs, which are files that contain information about calls made on telephone systems and are collected on a regular basis for processing into usage, capacity, performance and diagnostic reports¹. These CDRs contained every CAP and SIP signals sent or received by the network every 15 seconds. A day's worth of CAP and SIP data used an average of about 10GB uncompressed.

Another important part of the data collection was obtaining a list of all TSAN pools and the numbers they contained. This was done by scraping a table from a Truphone Confluence page that contained this information, and parsing it

¹Gartner's glossary entry for Call Detail Record: <https://www.gartner.com/en/information-technology/glossary/cdr-call-detail-recording>, Accessed 30/8/2022

into a dictionary that consisted of the key value pairs of "number":"pool name". A template data frame was also created with every pool name, the country it belonged to and how many numbers it contained. This template would later be used to facilitate calculating success rates per pool for every window in the detection mechanism, as well as serve as a boiler plate data frame for the score ranking in the healing mechanism.

There was some preprocessing work necessary to make the CDRs usable. The first step was selecting only the relevant columns of the dataset because the SIP and CAP CDRs contained 239 and 126 columns, respectively. The selected columns were "timestamp_utc", "calling_address", "called_address", "ftn" and "start_event", as these columns contained the information necessary for the detection process. It is important to note that the "ftn" column, which stands for "forward to number", only exists in the CAP CDRs.

After this, it was necessary to normalize the data by removing characters (i.e. the prefix "+") and cast the dataset to integer form to facilitate querying and filtering the dataset. Since the only records that were relevant were those that regarded TSAN usage, CDRs that contained a known TSAN number in the fields "called_address" or "ftn". There is also the need to anonymize numbers not part of Truphone's TSAN infrastructure. Additional information about which TSAN pool a number belonged to was also added to the CDR it belonged to.

Due to confidentiality and anonymity concerns, it is not possible to show real CDRs or post-processing data. However, a sample of the first and last ten entries in the CAP and SIP data generated by the simulator defined in Section 5.5.1 can be seen in Figures D.1 and D.2 of Appendix D, respectively. The phone numbers and timestamps are all randomly generated, and the pool names are fake. These datasets would correspond to the post-processing data.

5.2 Detection method

To understand the detection mechanism, one must first look at start of the usual TSAN call flow for CAP-enabled networks, described in subsection 4.4.3 and illustrated in Figure 4.7. The heuristic used in the detection method is based on this knowledge. A general architecture of the detection mechanism is presented in Figure E.1 of Appendix E.

There were several reasons for choosing a heuristic model over more advanced learning models, which are:

- The available data was very complex in structure, as many rows in a CDR file correspond to the same call, and the perceived success of a call was dependent on the presence or absence of certain rows. This structure made it very hard to find a suitable learning algorithm.
- The experts had already defined a heuristic model of what comprised a successful and unsuccessful call.

- Additionally, even if the data were to be restructured into a more palatable form, there was no labelled data to train a model with, and any labelling would either have to follow the model designed by the experts or involve collecting new data, which was dependant on more outages happening.

The proposed solution begins by initializing a table for a given time window with metrics for every pool. These metrics include the following:

- **Calls:** The amount of calls that were detected in that window. This is equivalent to the number of unique initial CAP IDPs present in that window.
- **Successes:** The amount of calls that were considered successful in that window.
- **Success Rate:** The ratio between successful calls and call attempts, calculated as:

$$SuccessRate(t) = \frac{SuccessfulCalls}{TotalCalls} \quad (5.1)$$

- **Retrigger Rate:** The ratio between call attempts and retriggers. This is calculated because it is the current methodology employed by the FrontOffice team to calculate success rates for pools, and it would be interesting to notice if there is any significant difference between including SIP Invites in the detection.
- **Failed SIPs:** The amount of calls that did not have a corresponding SIP Invite. Used for debugging.
- **Missed Retriggers:** The amount of calls that did not have a corresponding retrigger. Used for debugging.
- **Pool:** Information by the pool, formatted as "{pool centre abbreviation}.{pool location/name abbreviation}", e.g.: LDN.UK, AMS.DE, LDN.ES.
- **Pool country:** A two letter code for the country to which the pool is assigned. For example, the LDN.DE pool is assigned to Germany, so this field would be "DE".
- **Number count:** How many TSANs are assigned to each pool. This is later used for the healing mechanism, and kept in this table for ease of access.

The solution then takes CAP and SIP CDR aggregates them in time windows and selects the records for each window that are TSAN related. After this selection is made, a call flow check is made for every CAP IDP that has a TSAN number in the field "ftn". This starts by extracting to which pool the TSAN is from, and the call counter is incremented by one for that pool. Next, it searches for the corresponding SIP Invite and retrigger to that initial IDP. If both the SIP Invite and the retrigger are found, the successful calls counter for that pool is incremented by one. The SIP and retrigger counts are also incremented if they arrive within the schedule, regardless of whether or not the call is considered successful (i.e. if the

SIP is not found, but there is a retrigger, the retrigger count is incremented, and vice-versa).

After iterating every CAP IDP with a TSAN in the "ftn" field, the success rate is calculated. As defined by the network specialists, if the success rate is below 90%, an outage is declared.

This solution attempts to address most of the issues raised in Section 4.4.3. Its main advantage is the fact that it considers SIP failures, unlike the alerting mechanism used by the FrontOffice team which only uses IDPs and retriggers. It also produces additional metrics to evaluate what types of failures are occurring through the missed SIP and retrigger counters. Furthermore, it is impossible for this solution to detect more failures than call attempts, unlike the FrontOffice mechanism, making data analysis more reliable. It also uses the same pool identifiers used by the network, unlike the FrontOffice mechanism which looks at global title groups. While using global titles allows the engineers to better understand where the outage is occurring geographically, they are not useful for an automatic mitigation action.

5.3 Diagnostics method

As mentioned previously, if the success rate is below 90%, an outage is declared. This poses a problem because in the event that that window only includes for example five calls for a given pool and four of those calls fail, an outage is declared even though there is not reasonably enough evidence to justify declaring an outage and switching a pool. So, in the interest of clarifying pool statuses and avoiding unnecessary pool changes, if a pool did not have at least 20 calls for a given time window, the mechanism would look back to up to six hours to find call information that would allow to paint a clearer picture regarding whether or not the pool was failing. This number of calls and hours was defined as a good compromise between data abundance and relevance, because 20 calls were considered minimally satisfactory to understand a pool's status and six hours was short enough for the network's context to remain relevant. If the hour threshold is too great, the mechanism may look at periods whose context does not mirror that of the time of detection, e.g. there was an outage that had already passed and was no longer an issue.

In contrast, when network engineers face the same problem (insufficient information about the pool's health), they usually perform manual testing of the pool by "polling" the pool, i.e. making a batch of calls to numbers in that pool using a platform called SIGOS (refer to footnote 5 in section 4.4.3) to gain more information at the moment of detection. The amount of calls in this batch begins at 10 calls as standard, but additional batches of calls can be performed if the engineer does not find the results of the polls satisfactory. This showcases one of the limitations of automation versus manual work, because not only is it not possible to automate these tests without an Application Programming Interface (API), but also it is not trivial to automatize this process that is so heavily reliant on operator interpretation and knowledge.

In any case, it was defined that if the mechanism could not find 20 calls made in that pool up to 6 hours prior to the time window in question, it would disregard that pool as functioning normally. Nevertheless, if the system was to be implemented into production, it would warn the Network team flagging these pools as potential failures and leave further analysis to their discretion. This warning already exists, but the reporting mechanism would have to be defined.

5.4 Healing method

The healing mechanism's general flow is described in Figure F.1 of Appendix F. It uses an optimized version of the MCDM method WSM, also known as SAW (Yang [2014]). This model was chosen because it is a popular method due to its strength in single dimensional problems (as is the case), as well as simplicity and ease of use. This simplicity also ensures that it is easy for a network engineer not only to understand the results straightforwardly but also reconfigure the model's weights if they disagree with the results. A graphical representation of the adopted WSM is shown in Figure F.2 of Appendix F.

The criteria were also well-defined a priori by understanding what usually motivates the engineers' thought process when a pool needs to be changed. Some of these criteria are simple metrics, such as call counts and geographical distance, and other are compound, i.e they are derived from simple criteria, such as occupation and failure rates. Furthermore, since this method provides a ranking rather than a singular solution, the engineer may find it helpful to have several secondary choices if, for example, the first ranked pool is affected by some outside condition that makes it less than ideal for substitution. The criteria are the following:

- **Capacity:** There is a list with information about pools, including number counts per pool. Stands to reason that the more numbers there is in a pool, the more appealing it is as a substitute pool. It was considered to calculate capacity C using the count of numbers of the candidate pool (N_{new}) and the count of numbers in the affected pool (N_{old}) as in equation 5.2 in order to reward capacities above that of the affected pool and punished those below. This was quickly discarded because it added unnecessary complexity to the algorithm, opting instead by capacity C as calculated in equation 5.3.

$$C = \left(\frac{N_{new}}{N_{old}}\right)^2 \quad (5.2)$$

$$C = \frac{N_{new}}{N_{old}} \quad (5.3)$$

- **Occupation rate:** How many calls were made in the past detection window over how many numbers exist for that TSAN pool. Calls are counted by the detection mechanism by pool.

- **Failure rate:** How many calls failed in the past detection window over how many numbers exist for that TSAN pool. This is calculated as $FailureRate(t) = 1 - SuccessRate(t)$. $SuccessRate(t)$ is calculated by the detection mechanism.
- **Geographical Distance:** How far away is the evaluated pool from the malfunctioning pool. Since it is very difficult to determine exactly where each pool is located, a list with distances between country capitals was used with the assumption that most calls originate from there.
- **Protocol Binarization:** Since pools that use CAP rather than ISUP are preferred, if a pool uses ISUP it gets a value of 1 in this field, otherwise it gets a value of 0. This is due to the fact that lower scores are better, and thus having a preferred protocol should grant a lower score.
- **Price per call:** Prices per call are different in every pool. This is external data that was not able to be collected at the time of development so this criterion is left as future work.

To ensure fairness between attributes, they must be normalized. Both the z-score and the min-max methods of normalization were applied and their results can be compared in Tables 5.7 and 5.8, which are the results of a test of the mechanism for an outage in a UK pool.

The healing mechanism would need to generate a configuration to be uploaded to the REM Rhino node, signalling a change in pool configurations. The engineers need to be notified of this change to later effect a rollback.

5.5 Testing Plan

Testing the developed mechanism is divided in two parts: detection and healing. These components are divided into their own testing plans because they can be evaluated independently and because there was little added value in testing the mechanism end-to-end since its efficiency and performance is only as good as its weakest component.

5.5.1 Testing the detection mechanism

Testing the detection mechanism entailed collecting data for a period where pools had healthy behaviour and for periods of identified outages. Whereas collecting data for periods of normalcy did not pose a challenge, this was not the case for periods of outages because reports about TSAN outages were not very clear about when outages began or ended (or were mitigated) in the network. Rather, the associated JIRA tickets only reported when the outage was detected, and mitigation dates were somewhat unclear. A considered alternative approach was to search for periods of outage reported by the already existing mechanism in use by the

FrontOffice team with relative proximity to the periods reported in the outage's JIRA ticket, but still this proved unsatisfactory because of the issues mentioned in section 4.4.3.

Nevertheless, the collected data were SIP and CAP CDRs corresponding to a week (from May 15th, 2022, to May 21st, 2022) of presumed normal TSAN behaviour with no outages and the following outages:

- 5:00 AM to 4:00 PM, June 7th, 2022: Chile pools
- 2:00 PM to 4:30 PM, February 23rd, 2022: Portugal pools
- 1:00 PM to 6:00 PM, February 7th, 2022: USA pools

These pools were chosen because they were considered to have reliable JIRA tickets. It is also important to note that since these tests took place in a laptop with limited storage capacity and CDR data occupied at least 10GB per day collected, namely due SIP CDR data, there was a limitation to how much data could actually be collected for testing.

Two combinations of window size and offset were being tested:

- One hour window size and five minutes offset;
- One hour window size and one hour offset (corresponding to the configuration used in already existing mechanism being used by the FrontOffice team);

There are two classes in this problem: positive, which corresponds to normal behavior, and negative, which corresponds to outage behavior. To evaluate the detection mechanism's classifications, four metrics were chosen: Precision, Recall, F1-Score and MSCC. Precision and Recall were chosen because they are part of the F1 formula and it would be informative to understand the weight each had to the F1 Score. The F1-Score was chosen because it is a popular metric and it deals well with unbalanced data, which is the case. Finally, the MSCC was chosen because it is a good alternative to the F1-Score (Chicco and Jurman [2020]). Additionally, regression metrics such as MAE were considered due to the fact that missing a correct classification by 1% was better than missing by 50%, but this was ultimately not tested.

The assumptions made were that any pool at or above 90% success rate in a normal period consisted in a true positive classification whereas any pool below 90% success rate was considered a false negative.

For outage periods, it was harder to classify true negatives and false positives because not all pools were expected to be in an outage state. Instead, the number of pools in the affected country was the threshold of expected number of failing pools (designated the outage threshold), and thus only that amount of failing pools in any given window in an outage period was considered a true negative. If there were more pools failing, those would be considered false negatives. Conversely, if there were less pools failing than expected, that difference

would be considered the amount of false positives, and the pools considered normal above that threshold would be considered true positives. Summarily, the classification would go as follows:

- **True Positives:**
 - any pool at or above 90% success rate in a normal period
 - any pool at or above 90% success rate at or below the outage threshold in an outage period
- **True Negatives:**
 - any pool below 90% success rate below the outage threshold in an outage period
- **False Negatives:**
 - any pool below 90% success rate in a normal period
 - any pool below 90% success rate above the outage threshold in an outage period
- **False Positives:**
 - any pool at or above 90% success rate above the outage threshold in an outage period

A problem with this approach is that it drastically reduces the amount of true negatives and false positives possible, but this problem is unavoidable with the available data.

To address this imbalance issue, a simulator of TSAN pools based on the knowledge defined in Section 4.4.3 was created. This work was partly inspired by the necessity of a testing platform for the self-healing solution and partly inspired by Bakhtiyari et al. [2013] who propose a simulation of CDRs to test a billing platform. The amount of numbers to generate was chosen on the basis that there had to be enough unique numbers to avoid calls with the same numbers in short time spans. The pools were also assigned a different amount of numbers to replicate the variability in pool sizes that exists in real world data.

The success rates used to calculate how many healthy and failing calls should be in each window were generated from a beta distribution because it is bounded $\in [0, 1]$ and allows for a skewed distribution within these bounds. This could be used to represent a distribution of success rates that would allow for a similar number of healthy and outage success rates.

This simulator generated 1000 random normal phone numbers with 10 digits and no prefixes. It also generated 500 random TSAN numbers with 10 digits and no prefixes for three synthetic TSAN pools:

- Pool "A" was assigned 50 numbers (10%), all starting with a random digit $\in [1, 3]$

- Pool "B" was assigned 300 numbers (60%), all starting with a random digit $\in [4, 6]$
- Pool "C" was assigned 150 numbers (30%), all started with a random digit $\in [7, 9]$.

The beta distribution used to generate the success rates has parameters $\alpha = 18$ and $\beta = 2$. The `numpy.random` package for Python was used for this purpose. These parameters were chosen because they represent a distribution with mean $E[X] = \frac{\alpha}{\alpha + \beta} = \frac{18}{20} = 0.9$, which is the defined threshold between outage and healthy success rate values. This means that there is a statistical guarantee there will not be a great difference between the number of outage and healthy periods generated, granting balance to the resulting data set. The standard deviation and skewness of this distribution are $\sigma = SD(X) = 0.0655$ and $\gamma = 1.1109$, respectively. These success rates are later used as the real labels to evaluate if the system detects outages correctly.

For every time window, a random number of calls is generated for each pool, which is then stored in a matrix. By using the generated success rates and call count matrices, the simulator then calculates how many successful and failed calls should be generated, stores these amounts in separate matrices and generates the corresponding CAP IDPs, SIP Invites and retriggers for every time window. In case of a successful call, it generates the three signals correctly and in the correct time-frame. In case of a failed call, one of three cases may happen:

- A call flow with a missing retrigger is generated;
- A call flow with a missing SIP Invite is generated;
- A call flow with SIP Invites and retriggers arriving too late is generated

The CAP and SIP datasets are generated only containing the fields that would be present in a real world dataset after it was pre-processed to avoid unnecessary overhead.

Furthermore, if the window + offset combination chosen results in overlapping windows, the success rate matrix is recalculated using the generated call count and success count matrices to ensure that the labelling is accurate.

Finally, the detection mechanism is tested against the generated call flows, and the detected success rates are compared to the generated success rates using the Precision, Recall, F1 Score and MSCC metrics using the following rules:

- **True Positives:**
 - any pool at or above 90% success rate in a normal period
- **True Negatives:**
 - any pool below 90% success rate in an outage period

- **False Negatives:**
 - any pool below 90% success rate in a normal period
- **False Positives:**
 - any pool at or above 90% success rate in an outage period

Additionally, the difference between the detected success rates and the generated success rates is also used using the MAE to further understand how accurate the detection mechanism is in this synthetic context.

Seven days worth of data and the two window size + offset combinations tested on real data were tested in this data. The parameters for each simulation were the following:

- **One hour window size + one hour offset:** The number of calls generated per hour is never less than 50 per pool, with a maximum of 500 calls per hour.
- **One hour window size + five minute offset:** The number of calls generated every five minutes is never less than 50 per pool, with a maximum of 500 calls every five minutes.

5.5.2 Testing the healing mechanism

The healing mechanism was tested using an example outage detected by the detection mechanism. The chosen date for the outage was May 19th, 2022 at 3:00 AM in a UK-based pool. Whether or not the detection was accurate is not relevant for the purposes of the demonstration because the intention is evaluating if, in a given network state, the produced ranking made sense. This could not be formally tested because there usually are no absolute right answers for this problem, and thus there is no good way of measuring the accuracy of the healing mechanism. Nevertheless, the produced ranking was evaluated by experts to ascertain whether or not it was a valid assessment.

Two variations of the healing mechanism were tested: one using a min-max scaler to scale all features using minimum and maximum values and another using z-score normalization to normalize all features using mean and standard deviation. The weights used in these tests were the following:

- Distance weight = 0.4
- Occupation weight = 0.2
- Failure weight = 0.2
- Capacity weight = $-1 * 0.02$
- Protocol weight = 0.18

These weights were selected according to expert suggestion. While they did not suggest exact values, they suggested an order of priority, and the weights were adjusted according to the obtained results. The capacity weight was negative because the lower scores were more desirable, so greater capacities would need to lower the score. Additionally, all pools with a capacity between 0 and 1 were discarded.

5.6 Deployment and Integration Plan

As a more theoretical exercise, the potential implementation of this self-healing mechanism was considered. There are a number of constraints present to deploying the mechanism in the network, namely the fact that a direct connection between the healing mechanism and the REM node that would receive a new configuration is not possible. The proposed implementation is presented in Figure 5.1.

The plan, in a general sense, was to deploy the mechanism in a AWS Lambda function, where it would have access to the data stored in S3, and in the event of a healing action the configuration would have to be sent to an exposed API in a network element that would then be capable of relaying the configuration to the REM Rhino nodes.

Naturally, this plan could raise some security concerns that would need to be addressed, especially when it comes to exposing a network element through an API. These concerns, however, were beyond the scope of the research being conducted and were not explored.

5.7 Results

Before diving into the results of the whole mechanism, it is valuable to see the mechanism's progression. The first developed component was the detection mechanism, and initially it detected outages globally. The first tests were performed in a single day of normal behaviour using windows of one hour and offsets of 5 minutes to understand if the mechanism could correctly associated IDPs, SIP Invites and retriggers. Figure 5.2 presents how many signals of each type existed at every window of the day, as well as how many successful calls were detected for those windows. Figure 5.3 presents how many retriggers and SIP Invites were not correctly associated with their corresponding IDP. Figure 5.4 presents the success rate as a percentage for the whole day.

It is clear that the first prototype did not associate all the signals in the call pre-flow correctly, since the number of SIP Invites, retriggers and IDPs is roughly equal for the whole day but the number of detected successful calls is always lower than the number of calls (Figure 5.2). After inquiring network specialists and further analysis, revealed that this is possibly due to the fact that the prefix "C571" was added to the calling address in TSAN calls using the Germany pools,

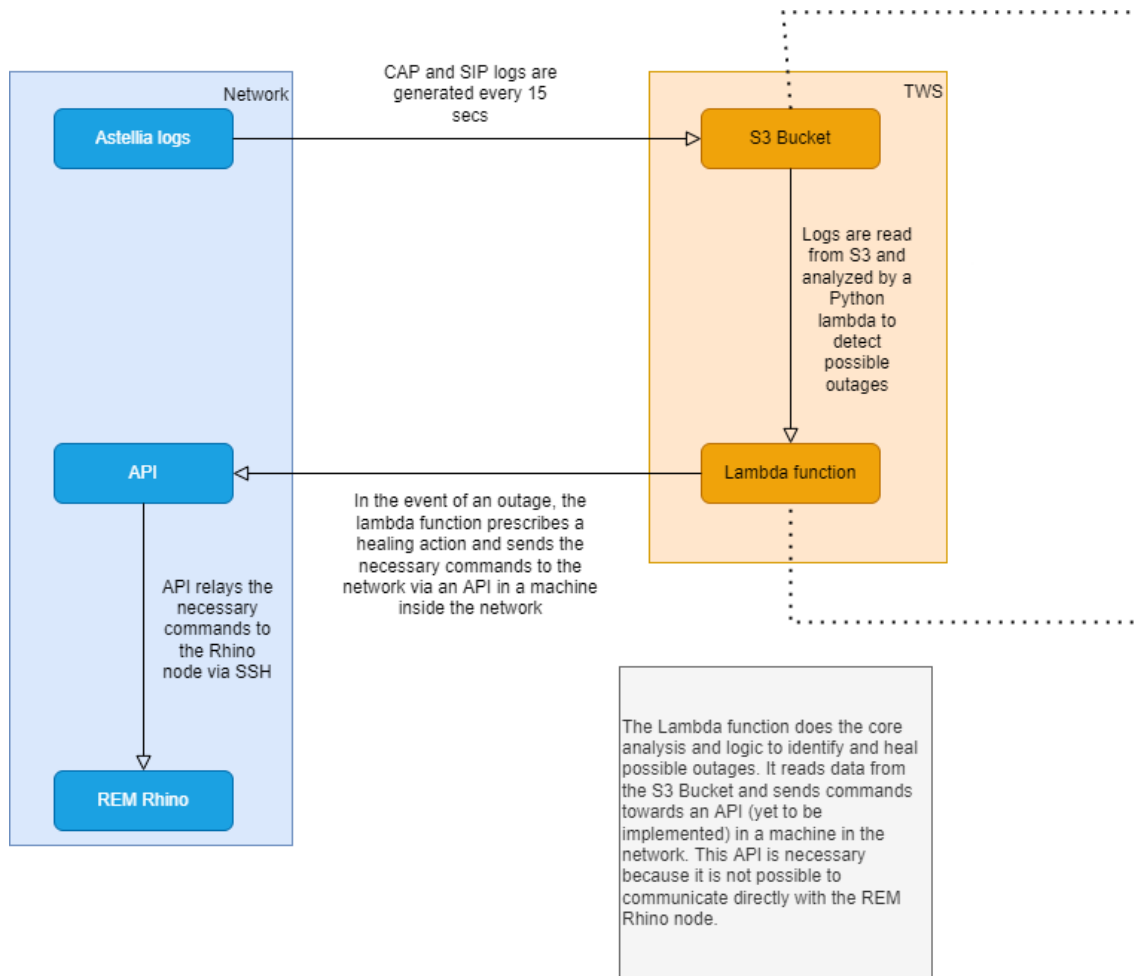


Figure 5.1: Proposed integration plan

i.e. "49 171 1234567" would be translated to "C571 49 171 1234567". This caused the calling address in the IDP did not correspond to the calling address in the retrigger and disrupted the process of association of corresponding signals in the developed mechanism.

Nevertheless, the fact that the number of unattributed SIP Invites (Figure 5.3) is close to 0 suggests that, despite this slight setback, the mechanism was detecting call flows appropriately, although Figure 5.4 may suggest otherwise since there is a large portion of the day with a success rate below 90%. Another interesting takeaway from this experiment is that it is not uncommon for there to be more retriggers than IDPs. The reason for this is unknown, but it is not detrimental to the mechanism's behavior.

The second prototype was evaluated more formally by following the test plan detailed in section 5.5. Regarding the detection mechanism, the results for the one hour window size + five minute offset and the one hour window size + one hour offset tests are presented as confusion matrices in Tables 5.1 and 5.2 respectively, whereas the overall results of the metrics used to evaluate the performances obtained are presented in Table 5.3. It is important to note that only pools that had a non-negative success rate were considered. This was because when a pool had no calls for a given hour, the mechanism gave it a success rate

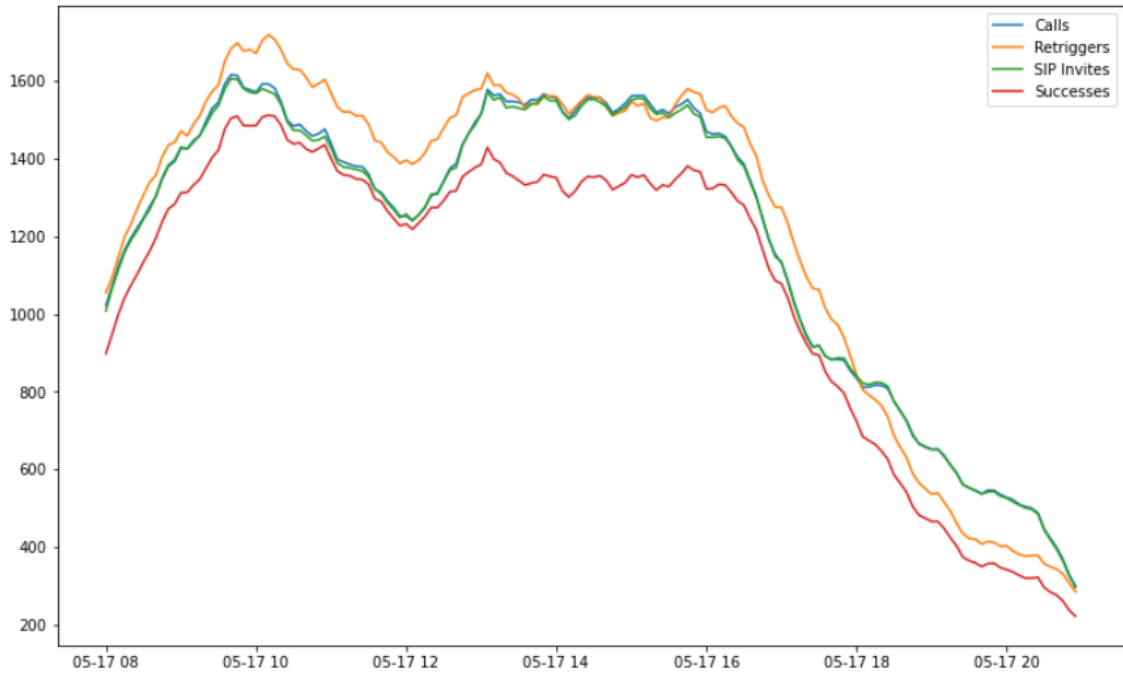


Figure 5.2: Comparison of existing signals and detected successes

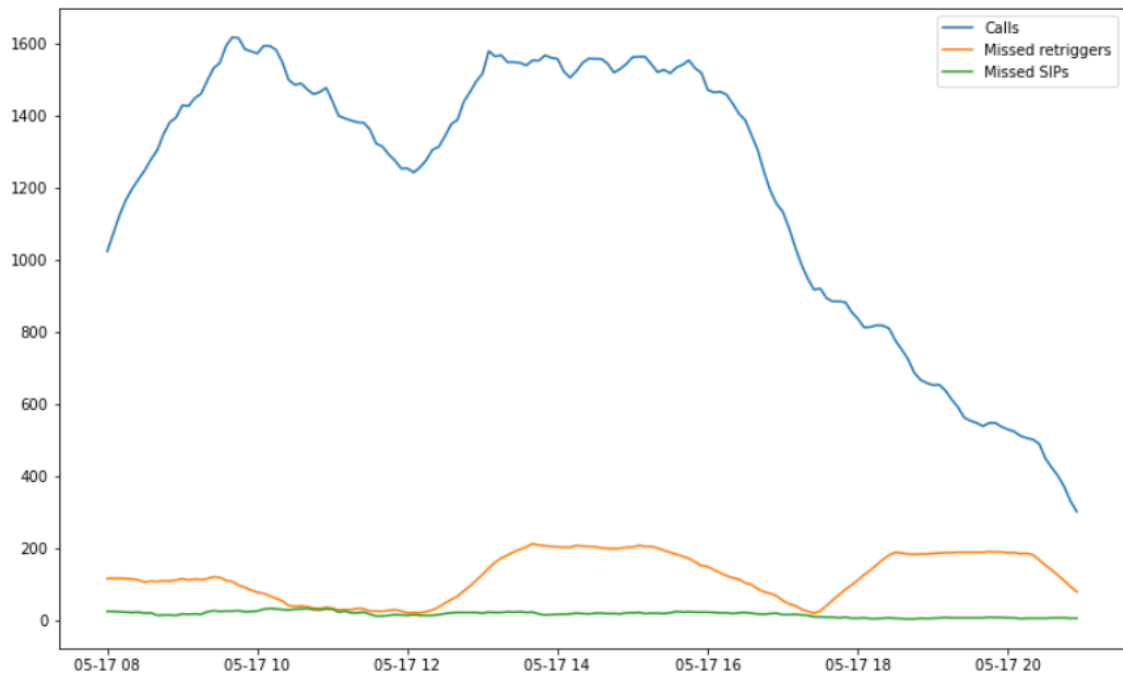


Figure 5.3: Failed retriggers and SIP Invites

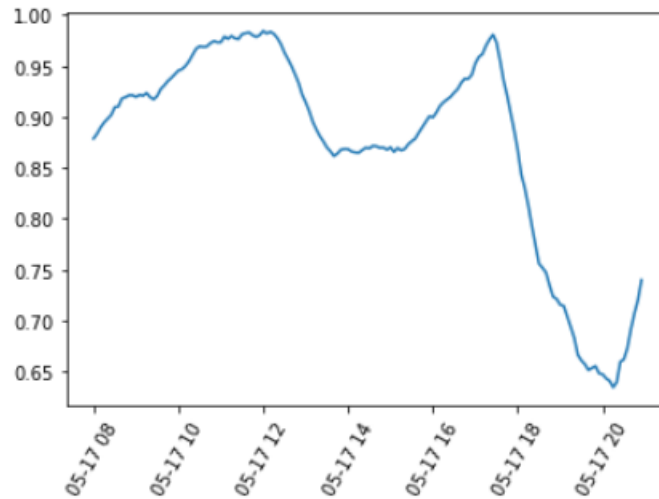


Figure 5.4: Success rate as a percentage

of -0.1 precisely to mark it as non-informative.

		True classes		Total
		Positive	Negative	
Detected classes	Positive	8330	731	9061
	Negative	524	56	580
Total		8854	787	9641

Table 5.1: Confusion matrix of the one hour window size + five minute offset detection tests using real data

		True classes		Total
		Positive	Negative	
Detected classes	Positive	727	40	767
	Negative	60	7	67
Total		787	47	834

Table 5.2: Confusion matrix of the one hour window size + one hour offset detection tests using real data

	One hour window + five minutes offset	One hour window + one hour offset
Precision	0.9408	0.9478
Recall	0.9193	0.9237
F1	0.9299	0.9356
MSCC	0.0112	0.0516

Table 5.3: Test results for the detection mechanism using real data

The precision, recall and F1 metrics suggest that the mechanism is relatively competent at identifying periods of normal behaviour, but the MSCC suggests that the mechanism may not be as capable of identifying outages. These results may be caused by the lack of properly labelled data.

The overall results using simulated data are presented in Table 5.6. The resulting confusion matrices for the one hour window size + one hour offset and the results for the one hour window size + five minute offset are presented in Tables 5.5 and 5.4, respectively.

		True classes		Total
		Positive	Negative	
Detected classes	Positive	245	0	245
	Negative	3	256	259
Total		248	256	504

Table 5.4: Confusion matrix of the one hour window size + one hour offset detection tests using simulated data

		True classes		Total
		Positive	Negative	
Detected classes	Positive	989	0	989
	Negative	38	989	1027
Total		1027	989	2016

Table 5.5: Confusion matrix of the one hour window size + five minute offset detection tests using simulated data

	One hour window + five minutes offset	One hour window + one hour offset
Precision	1.0	1.0
Recall	0.9629	0.9879
F1	0.9811	0.9939
MSCC	0.9629	0.9884
MAE	0.0123	0.0057

Table 5.6: Test results for the detection mechanism using simulated data

These results show that the mechanism is very accurate at correctly detecting the simulated outages and healthy periods. The slight inaccuracy may be due to the fact that healthy calls on the edges of the windows may have the CAP IDP in one window, and the corresponding SIP Invite and/or retrigger in the following window. The calculated MAEs are also very low and indicate that the inaccuracy of the five minute offset variant is slightly higher than that of the one hour offset variant.

As for the healing mechanism, the rankings of the z-score and min-max variations are represented in Tables 5.7 and 5.8, respectively. An interesting conclusion is that the min-max variant gives preference to pools whose distance is smaller, whereas the z-score variant gives preference to pools with a greater capacity. This is most likely due to the fact that there are outliers farther away from the observed distribution in the capacity feature set than in the distance feature set, and thus they are attributed a greater value using the z-score normalization.

Regarding expert validation, the presented choices seemed to make more sense in the min-max variation than in the z-score variation purely because they came closer to the decision engineers would make in this scenario.

Pool Name	Country	Distance	Calls	FRate	Protocol	Capacity	Occ.	Score
LDN.AT	AT	1211	0	0	CAP	49.6	0	-10.5777
LDN.UK	GB	0	0	0	CAP	20.0	0	-9.8110
AMS.UK	GB	0	0	0	CAP	20.0	0	-9.8110
AMS.EU_UK	GB	0	0	0	CAP	4.4	0	-8.1278
LDN.EU_UK	GB	0	0	0	CAP	4.4	0	-8.1278
LDN.UK_ISR	GB	313	0	0	CAP	4.3	0	-8.1170
AMS.UK_ISR	GB	313	0	0	CAP	4.3	0	-8.1170
LDN.NL_KPN	NL	365	0	0	CAP	9.9	0	-8.0662
AMS.NL_KPN	NL	449	0	0	CAP	9.9	0	-8.0662
AMS.FR_ORANGE	FR	496	0	0	CAP	4.9	0	-7.4786

Table 5.7: Results of the healing mechanism using Z-Score normalization

Pool Name	Country	Distance	Calls	FRate	Protocol	Capacity	Occ.	Score
LDN.UK	GB	0	0	0	CAP	20.0	0	-0.2010
AMS.UK	GB	0	0	0	CAP	20.0	0	-0.2010
AMS.EU_UK	GB	0	0	0	CAP	4.4	0	-0.0434
LDN.EU_UK	GB	0	0	0	CAP	4.4	0	-0.0434
LDN.UK_ISR	GB	0	0	0	CAP	4.3	0	-0.0424
AMS.UK_ISR	GB	0	0	0	CAP	4.3	0	-0.0424
LDN.AT	AT	1211	0	0	CAP	49.6	0	0.0148
LDN.IMS_UK	GB	0	1	0	CAP	3.4	0.02941	0.0440
AMS.IMS_UK	GB	0	1	0	CAP	3.4	0.02941	0.0440
AMS.NL_KPN	NL	340	0	0	CAP	9.9	0	0.0455

Table 5.8: Results of the healing mechanism using Min-Max scaling

Chapter 6

Conclusion

This document details research about the topic of self-healing of outages in a MVNO context, from the study of the existent outage categories to the research and development of a self-healing solution for a particular outage event type. This topic is rather complex and involved extensive research and knowledge acquisition about many different technologies and fields, which were summarily described in the appropriate order.

It begins by exploring the relevant technological concepts related with MVNOs, and how they are at the forefront of mobile network innovation, partly due to their inherent dependence on technologies such as NFV and SDN. These technologies allow networks to rely less on hardware focused architectures and more on modern software engineering principles and decentralized logic and management, making networks that have implemented them much more versatile and flexible. This versatility and flexibility may provide the necessary conditions to solve network outages with more sophisticated programmatic procedures and approaches, and further advance this field of research.

Nevertheless, the flexibility of software engineering principles is also accompanied by common issues in software deployments. This and the fact that networking and connectivity services have become a great necessity rather than a luxury, like many other communication technologies before them, contribute to the also ever growing impact of outages. As explored in subsection 2.2.3, millions of dollars are lost per year to outages of varying types and causes, so it is necessary to further improve our knowledge and solutions for these outages. While many have proposed numerous different methods of evaluating and handling the many problems that can arise in networks, there is still much work to be done especially when it comes to organizing and standardizing these solutions.

This document naturally pays most attention to those solutions that can be integrated in self-healing mechanisms, either in outage detection, diagnosis or compensation/recovery capacities. These mechanisms have been studied for over two decades for the cellular domain of mobile networks, but the relative recency of the aforementioned NFV and SDN technologies has not yet provided the research community the opportunity to produce such an extensive and exhaustive set of approaches for all three components combined.

Truphone's outage history and methodology provided insights into the difficulties of automating outage handling processes via self-healing. The analyzed outages tended to be very specific in nature and, even though categorizing them is not only a possibility but also a necessity, outages in the same categories can be vastly different to the point some can be subjects of self-healing whereas others cannot. Nevertheless, a well-structured outage categorization methodology, resorting to tools such as priority matrices, goes a long way to make outage histories easier to understand and analyze.

Unsurprisingly, good communication is also key to resolving existing outages. A proof of this was the first outage candidate. As explained in section 4.4.2, a direct result of simply communicating preliminary research of an outage that disrupted services in Australia to the appropriate Truphone personnel was the resolution of said outage.

The developed self-healing mechanism also supports this conclusion because it was heavily reliant on the knowledge of experts. Much of the time spent in research and development was spent understanding the underlying technologies and concepts, how they interacted and what failure in these interactions generated an outage. And even during and after the development of both prototypes, there were many small constraints and caveats that appeared and changed the way the mechanism had to operate, some more drastically than others.

Due to the complexity of the SDN paradigm, particularly in Truphone, it is safe to say that developing and deploying a fully functional self-healing mechanism is a task that requires a deep understanding of all associated technologies.

However, the results showed that even this minimally viable prototype was capable of generating both insights and automatically performing tasks that would otherwise needed manual labor, even if it is far from being deployable. It also showed a potential alternative to existing mechanisms, its strengths and drawbacks.

Briefly, the contributions of this work were:

- A state of the art survey regarding outages, mobile networks and self-healing relevant to the proposed solution.
- A categorization of Truphone outages, based on existing literature and on the analysis of Truphone data, and an attempt at aligning this categorization with other categorizations found in the state of the art.
- The well-documented iterative development of a prototype self-healing system in a MVNO context that is able to detect, diagnose and compensate/recover from an outage that affects calls that require use of a TSAN (see Section 4.4.3), with all major design decisions well justified. If applied in production, this system saves man hours and increases the customer experience, by recovering from the target outage faster.
- An assessment of the impact that this prototype may have on the business, as well as a suggestion of future work (see Section 6.1).

- A better and deeper understanding about the TSAN outage at Truphone.
- As a result of the discovery process of the appropriate type of outage to be targeted by the self healing process, other issues that were causing outages were detected, and as a consequence of this an issue was fixed resulting in Truphone saving resources by not having to mitigate the outage it caused.
- A simulator of TSAN-related call records, capable of generating labeled data according to a predefined beta distribution of success rates to create a balanced synthetic dataset.

6.1 Future work

There is much work that can still be done to both improve the understanding of outages in an MVNO and iterate upon the self-healing prototype developed. Some of that possible future work is presented in this section.

Automatic categorization of outages based on a text mining or NLP approach

During the outage categorization research, a brief experiment of automatic outage categorization based on JIRA tickets' descriptions was performed. Although this avenue of research was only briefly explored, it generated interesting results worthy of further exploration. The more in-depth exploration of TF-IDF and LDA configurations could yield better results

The use of more refined approaches, namely based on NLP, such as attempting to extract relevant chunks and/or phrases from the descriptions or training neural networks such as the Long Short-Term Memory (Lstm) Neural Network or even Transformer-based classifiers such as BERT (Devlin et al. [2018]) for text classification, could also perhaps result in a valuable tool for analyzing Truphone's outage history.

Better streamlined data collection for the prototype

One of the limitations in the development of this prototype was the limitations in storage and processing power of working from a personal computer. Moving the mechanism to a cloud service such as AWS is the logical solution to this problem. This transition, however, is not without its challenges. The solution would have to be completely adapted to function in a Lambda function and it would need read privileges to the existing S3 bucket, which requires a higher level of scrutiny of the quality and safety of the processes and components used in this solution.

Considering ISUP networks in the developed prototype

Another limitation in the developed prototype was the different detection rule set necessary to detect successful calls in ISUP networks such as those in Spain and Hong Kong. Adding these countries' networks to the fold would increase data efficiency because more pools could be considered. Additionally, this is a requirement for the mechanism is to be eligible for deployment because it must monitor all of Truphone's TSAN pools.

Properly parsing prefixes in the developed prototype

There are many networks that modify phone numbers, for example translating the number from a national format to its international E.164 format e.g., 912345678 becomes +351912345678 if a Portuguese caller tries to call a foreign network. There are also cases where Truphone's network adds prefixes when numbers try to call international networks, such as Germany (the added prefix is "C571"). These changes deeply affected the detection mechanism's efficacy and need to be fully considered to make the mechanism more robust, versatile and stable.

Developing and testing forecasting of the TSAN outage

An interesting avenue of research would be the research and development of a forecasting mechanism for the TSAN outage, perhaps based on the calculated success rates, call counts, retrigger counts and SIP Invite counts generated by the detection mechanism. It should be noted that network engineers expressed some cynicism on the efficacy of such a mechanism because they feel it is a particularly hard problem to predict. Nevertheless, the insights produced by this approach could be valuable in at least further understanding the boundaries of self-healing in MVNOs.

Creating an improved staging environment for the TSAN outage

Despite having simulated TSAN call flows according to the existing knowledge (see Section 5.5.1), having a staging environment capable of simulating the network in the event of a TSAN outage would be invaluable. The main objectives of such a staging environment would be synthesizing data with real world characteristics and testing the mechanism in action without having to wait for real world conditions to be appropriate for testing.

Labelling data for further testing of the mechanism

One of the greatest difficulties in testing the mechanism was obtaining labeled data that was reliable. Having a properly curated dataset of periods of both nor-

mal and outage behaviour would greatly improve the quality of test results and, consequently, the assessment of any iteration of the prototype.

Test for more window sizes and offsets of the detection mechanism

To find the optimal configuration of the detection mechanism, it is imperative that the ideal window size and offset is found. This can be done through a grid search of these two parameters, but it is a computationally intensive task to cast a wide net.

Integrating the mechanism in the network and evaluating its performance

Naturally, deploying the mechanism on the network and having experts make use of it and compare it to the existing outage handling methods would provide valuable insights into its viability and value. For this to be a possibility, however, the prototype would have to be more robust by addressing some of the aforementioned work.

References

- Giuseppe Aceto, Alessio Botta, Pietro Marchetta, Valerio Persico, and Antonio Pescapé. A comprehensive survey on internet outages. *Journal of Network and Computer Applications*, 113:36–63, 2018.
- Pankaj K Agarwal, Alon Efrat, Shashidhara K Ganjugunte, David Hay, Swaminathan Sankararaman, and Gil Zussman. Network vulnerability to single, multiple, and probabilistic physical attacks. In *2010-MILCOM 2010 MILITARY COMMUNICATIONS CONFERENCE*, pages 1824–1829. IEEE, 2010.
- Shikha Agrawal and Jitendra Agrawal. Survey on anomaly detection using data mining techniques. *Procedia Computer Science*, 60:708–713, 2015. ISSN 1877-0509. doi: <https://doi.org/10.1016/j.procs.2015.08.220>. URL <https://www.sciencedirect.com/science/article/pii/S1877050915023479>. Knowledge-Based and Intelligent Information & Engineering Systems 19th Annual Conference, KES-2015, Singapore, September 2015 Proceedings.
- Mohiuddin Ahmed, Abdun Naser Mahmood, and Md Rafiqul Islam. A survey of anomaly detection techniques in financial domain. *Future Generation Computer Systems*, 55:278–288, 2016.
- Janne Ali-Tolppa, Szilard Kocsis, Benedek Schultz, Levente Bodrog, and Marton Kajo. Self-healing and resilience in future 5g cognitive autonomous networks. In *2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K)*, pages 1–8. IEEE, 2018.
- Mehdi Amirijoo, Ljupco Jorguseski, T Kurner, Remco Litjens, Michaela Neuland, Lars-Christoph Schmelz, and U Turke. Cell outage management in lte networks. In *2009 6th International Symposium on Wireless Communication Systems*, pages 600–604. IEEE, 2009.
- Martin Aruldoss, T Miranda Lakshmi, and V Prasanna Venkatesan. A survey on multi criteria decision making methods and its applications. *American Journal of Information Systems*, 1(1):31–43, 2013.
- Ahmad Asghar, Hasan Farooq, and Ali Imran. Self-healing in emerging cellular networks: review, challenges, and research directions. *IEEE Communications Surveys & Tutorials*, 20(3):1682–1709, 2018.
- Kevin Ashton et al. That ‘internet of things’ thing. *RFID journal*, 22(7):97–114, 2009.

- Alberto Avritzer, Domenico Cotroneo, Yennun Huang, and Kishor Trivedi. Software aging and rejuvenation: A genesis. In *Handbook Of Software Aging And Rejuvenation: Fundamentals, Methods, Applications, And Future Directions*, pages 3–19. World Scientific, 2020.
- Hanane Aznaoui, Arif Ullah, Said Raghay, Layla Aziz, and Mubashir Hayat Khan. An efficient gaf routing protocol using an optimized weighted sum model in wsn. *Indonesian Journal of Electrical Engineering and Computer Science*, 22(1):396–406, 2021.
- Prafulla Bafna, Dhanya Pramod, and Anagha Vaidya. Document clustering: Tf-idf approach. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 61–66. IEEE, 2016.
- Rostamzadeh Bakhtiyari et al. Developing a simulator application to test the billing platform in telecom industry. In *International Conference on Data Engineering (ICDE 2013)*, 2013.
- Marc Balon and Bernard Liau. Mobile virtual network operator. In *2012 15th International Telecommunications Network Strategy and Planning Symposium (NETWORKS)*, pages 1–6, 2012. doi: 10.1109/NETWKS.2012.6381694.
- Karyn Benson, Alberto Dainotti, Kimberly C Claffy, and Emile Aben. Gaining insight into as-level outages through analysis of internet background radiation. In *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 447–452. IEEE, 2013.
- David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.
- Internet Without Borders. Censure internet archives). <https://internetwithoutborders.org/category/censure-internet/>, 2021. Accessed: 2021-12-21.
- Álvaro Brandón, Marc Solé, Alberto Huélamo, David Solans, María S. Pérez, and Victor Muntés-Mulero. Graph-based root cause analysis for service-oriented and microservice architectures. *Journal of Systems and Software*, 159:110432, 2020. ISSN 0164-1212. doi: <https://doi.org/10.1016/j.jss.2019.110432>. URL <https://www.sciencedirect.com/science/article/pii/S0164121219302067>.
- Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.
- Krzysztof Cabaj, Jacek Wytrębowicz, Sławomir Kuklinski, Paweł Radziszewski, and Khoa Dinh. Sdn architecture impact on network security. 09 2014. doi: 10.15439/2014F473.
- Matteo Calabrese, Martin Cimmino, Francesca Fiume, Martina Manfrin, Luca Romeo, Silvia Ceccacci, Marina Paolanti, Giuseppe Toscano, Giovanni Ciandrini, Alberto Carrotta, Maura Mengoni, Emanuele Frontoni, and Dimos Kapetis. Sophia: An event-based iot and machine learning architecture for

- predictive maintenance in industry 4.0. *Information*, 11(4), 2020. ISSN 2078-2489. doi: 10.3390/info11040202. URL <https://www.mdpi.com/2078-2489/11/4/202>.
- Sinan Çalışır and Meltem Kurt Pehlivanoglu. Model-free reinforcement learning algorithms: A survey. In *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4. IEEE, 2019.
- Christine P Chai. Word distinctivity-quantifying improvement of topic modeling results from n-gramming. *REVSTAT-Statistical Journal*, 20(2):199–220, 2022.
- Tianfeng Chai and Roland R Draxler. Root mean square error (rmse) or mean absolute error (mae). *Geoscientific Model Development Discussions*, 7(1):1525–1534, 2014.
- Yujun Chen, Xian Yang, Qingwei Lin, Hongyu Zhang, Feng Gao, Zhangwei Xu, Yingnong Dang, Dongmei Zhang, Hang Dong, Yong Xu, et al. Outage prediction and diagnosis for cloud service systems. In *The World Wide Web Conference*, pages 2659–2665, 2019.
- Sergey Chernov, Michael Cochez, and Tapani Ristaniemi. Anomaly detection algorithms for the sleeping cell detection in lte networks. In *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, pages 1–5. IEEE, 2015.
- Davide Chicco and Giuseppe Jurman. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13, 2020.
- Kenjiro Cho, Cristel Pelsser, Randy Bush, and Youngjoon Won. The japan earthquake: the impact on traffic and routing observed by a local isp. In *Proceedings of the Special Workshop on Internet and Disasters*, pages 1–8, 2011.
- Rishabh Choudhary and Hemant Kumar Gianey. Comprehensive review on supervised machine learning algorithms. In *2017 International Conference on Machine Learning and Data Science (MLDS)*, pages 37–43, 2017. doi: 10.1109/MLDS.2017.11.
- Gabriela Ciocarlie, Ulf Lindqvist, Kenneth Nitz, Szabolcs Nováczki, and Henning Sanneck. On the feasibility of deploying cell anomaly detection in operational cellular networks. In *2014 IEEE Network Operations and Management Symposium (NOMS)*, pages 1–6. IEEE, 2014a.
- Gabriela F Ciocarlie, Christopher Connolly, Chih-Chieh Cheng, Ulf Lindqvist, Szabolcs Nováczki, Henning Sanneck, and Muhammad Naseer-ul Islam. Anomaly detection and diagnosis for automatic radio network verification. In *International Conference on Mobile Networks and Management*, pages 163–176. Springer, 2014b.
- Kimberly Claffy, Young Hyun, Ken Keys, Marina Fomenkov, and Dmitri Krioukov. Internet mapping: from art to science. In *2009 Cybersecurity Applications & Technology Conference for Homeland Security*, pages 205–211. IEEE, 2009.

- Massimo Condoluci and Toktam Mahmoodi. Softwarization and virtualization in 5g mobile networks: Benefits, trends and challenges. *Computer Networks*, 146:65–84, 2018. ISSN 1389-1286. doi: <https://doi.org/10.1016/j.comnet.2018.09.005>. URL <https://www.sciencedirect.com/science/article/pii/S1389128618302500>.
- United States. Congress. *Dodd-Frank Wall Street Reform and Consumer Protection Act: Conference Report (to Accompany HR 4173)*., volume 111. US Government Printing Office, 2010.
- Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.
- Alberto Dainotti, Claudio Squarcella, Emile Aben, Kimberly C Claffy, Marco Chiesa, Michele Russo, and Antonio Pescapé. Analysis of country-wide internet outages caused by censorship. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 1–18, 2011.
- Alberto Dainotti, Roman Amman, Emile Aben, and Kimberly C Claffy. Extracting benefit from harm: using malware pollution to analyze the impact of political and geophysical events on the internet. *ACM SIGCOMM Computer Communication Review*, 42(1):31–39, 2012.
- Shivani Deshpande, Marina Thottan, Tin Kam Ho, and Biplab Sikdar. An online mechanism for bgp instability detection and analysis. *IEEE transactions on Computers*, 58(11):1470–1484, 2009.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- Mentari Djatmiko, Dominik Schatzmann, Xenofontas Dimitropoulos, Arik Friedman, and Roksana Boreli. Collaborative network outage troubleshooting with secure multiparty computation. *IEEE Communications Magazine*, 51(11):78–84, 2013. doi: 10.1109/MCOM.2013.6658656.
- Patrick Donegan. Mobile network outages & service degradations: A heavy reading survey analysis. *Firmenschrift. Heavy Reading*, 2013.
- Patrick Donegan. Mobile network outages & service degradations: A heavy reading survey analysis. *Firmenschrift. Heavy Reading*, 2016.
- Benoit Donnet and Timur Friedman. Internet topology discovery: a survey. *IEEE Communications Surveys & Tutorials*, 9(4):56–69, 2007.
- Nick Duffield. Network tomography of binary network performance characteristics. *IEEE Transactions on Information Theory*, 52(12):5373–5388, 2006.
- Adel Eesa and Wahab Arabo. A normalization methods for backpropagation: A comparative study. *Science Journal of University of Zakho*, 5(4):319–323, Dec. 2017. doi: 10.25271/2017.5.4.381. URL <http://sjuoz.uoz.edu.krd/index.php/sjuoz/article/view/440>.

- C. Elliott and B. Heile. Self-organizing, self-healing wireless networks. In *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*, volume 1, pages 149–156 vol.1, 2000a. doi: 10.1109/AERO.2000.879383.
- C. Elliott and B. Heile. Self-organizing, self-healing wireless networks. In *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*, volume 1, pages 149–156 vol.1, 2000b. doi: 10.1109/AERO.2000.879383.
- Brian Eriksson, Ramakrishnan Durairajan, and Paul Barford. Riskroute: A framework for mitigating network outage threats. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, pages 405–416, 2013.
- NFV ETSI. Architectural framework, etsi gs nfv 002 v1. 2.1. *European Telecommunications Standards Institute*, 2013.
- Sujuan Feng and Eiko Seidel. Self-organizing networks (son) in 3gpp long term evolution. *Nomor Research GmbH, Munich, Germany*, 20:1–15, 2008.
- Eduard Glatz and Xenofontas Dimitropoulos. Classifying internet one-way traffic. In *Proceedings of the 2012 Internet Measurement Conference*, pages 37–50, 2012.
- Jose Manuel Navarro Gonzalez, Javier Andion Jimenez, Juan Carlos Duenas Lopez, and Hugo A. Parada G. Root cause analysis of network failures using machine learning and summarization techniques. *IEEE Communications Magazine*, 55(9):126–131, 2017. doi: 10.1109/MCOM.2017.1700066.
- Haryadi S. Gunawi, Mingzhe Hao, Riza O. Suminto, Agung Laksono, Anang D. Satria, Jeffry Adityatama, and Kurnia J. Eliazar. Why does the cloud stop computing? lessons from hundreds of service outages. New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450345255. doi: 10.1145/2987550.2987583. URL <https://doi.org/10.1145/2987550.2987583>.
- Xifeng Guo, Xinwang Liu, En Zhu, and Jianping Yin. Deep clustering with convolutional autoencoders. In *International conference on neural information processing*, pages 373–382. Springer, 2017.
- Vijay K Gurbani, Dan Kushnir, Veena Mendiratta, Chitra Phadke, Eric Falk, and Radu State. Detecting and predicting outages in mobile networks with log data. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2017.
- Bo Han, Vijay Gopalakrishnan, Lusheng Ji, and Seungjoon Lee. Network function virtualization: Challenges and opportunities for innovations. *IEEE Communications Magazine*, 53(2):90–97, 2015. doi: 10.1109/MCOM.2015.7045396.
- Yasuhiro Harada, Wang Hui, Yukinobu Fukushima, and Tokumi Yokohira. A reroute method to recover fast from network failure. In *2014 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 903–908, 2014. doi: 10.1109/ICTC.2014.6983329.

- Steffen Herbold, Alexander Trautsch, and Fabian Trautsch. On the feasibility of automated prediction of bug and non-bug issues. *Empirical Software Engineering*, 25:1–37, 11 2020. doi: 10.1007/s10664-020-09885-w.
- John B Horrigan. *Broadband adoption and use in America*. Federal Communications Commission Washington, DC, 2010.
- Mohammad Hossin and Md Nasir Sulaiman. A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process*, 5(2):1, 2015.
- Daiki Imahama, Yukinobu Fukushima, and Tokumi Yokohira. A reroute method using multiple routing configurations for fast ip network recovery. In *2013 19th Asia-Pacific Conference on Communications (APCC)*, pages 433–438. IEEE, 2013.
- Brookings Institution. <https://www.brookings.edu/search/?s=internet+outage+censorship>, 2021. Accessed: 2021-12-21.
- Shashwat Jain, Manish Khandelwal, Ashutosh Katkar, and Joseph Nygate. Applying big data technologies to manage qos in an sdn. In *2016 12th International Conference on Network and Service Management (CNSM)*, pages 302–306. IEEE, 2016.
- Umar Javed, Italo Cunha, David Choffnes, Ethan Katz-Bassett, Thomas Anderson, and Arvind Krishnamurthy. Poiroot: Investigating the root cause of inter-domain path changes. *ACM SIGCOMM Computer Communication Review*, 43(4): 183–194, 2013.
- Hamed Jelodar, Yongli Wang, Chi Yuan, Xia Feng, Xiahui Jiang, Yanchao Li, and Liang Zhao. Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications*, 78(11):15169–15211, 2019.
- MZH Jesmeen, J Hossen, S Sayeed, CK Ho, K Tawsif, Armanur Rahman, and E Arif. A survey on cleaning dirty data using machine learning paradigm for big data analytics. *Indonesian Journal of Electrical Engineering and Computer Science*, 10(3):1234–1243, 2018.
- Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 1972.
- Sunil Kappal et al. Data normalization using median median absolute deviation mmad based z-score for robust predictions vs. min–max normalization. *London Journal of Research in Science: Natural and Formal*, 2019.
- Ilyas Alper Karatepe and Engin Zeydan. Anomaly detection in cellular network data using big data analytics. In *European Wireless 2014; 20th European Wireless Conference*, pages 1–5. VDE, 2014.
- Ethan Katz-Bassett, Harsha V Madhyastha, John P John, Arvind Krishnamurthy, David Wetherall, and Thomas E Anderson. Studying black holes in the internet with hubble. In *NSDI*, volume 8, pages 247–262, 2008.

- Zohaib Khan, Naokhaiz Afaqui, and Osama Humayun. Detection and prevention of ddos attacks on software defined networks controllers for smart grid. *International Journal of Computer Applications*, 181:16–21, 03 2019. doi: 10.5120/ijca2019918572.
- Rana M Khanafer, Beatriz Solana, Jordi Triola, Raquel Barco, Lars Moltsen, Zwi Altman, and Pedro Lazaro. Automated diagnosis for umts networks using bayesian network approach. *IEEE Transactions on vehicular technology*, 57(4): 2451–2461, 2008.
- Diego Kreutz, Fernando M. V. Ramos, Paulo Esteves Veríssimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, 2015. doi: 10.1109/JPROC.2014.2371999.
- Maciej Kuźniar, Peter Perešini, Nedeljko Vasić, Marco Canini, and Dejan Kostić. Automatic failure recovery for software-defined networks. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, pages 159–160, 2013.
- Jukka Lehtikainen, Pierre Pont, and Yannick Sent. Virtually mobile: What drives mvno success. *Telecom, Media & High Tech Extranet*, 2014.
- Asma Ben Letaifa. Adaptive qoe monitoring architecture in sdn networks: Video streaming services case. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1383–1388. IEEE, 2017.
- Jun Li and Scott Brooks. I-seismograph: Observing and measuring internet earthquakes. In *2011 Proceedings IEEE INFOCOM*, pages 2624–2632. IEEE, 2011.
- Zheng Li, Mingfei Liang, Liam O’Brien, and He Zhang. The cloud’s cloudy moment: A systematic survey of public cloud service outage. *arXiv preprint arXiv:1312.6485*, 2013.
- Qi Liao, Marcin Wiczanowski, and Sławomir Stańczak. Toward cell outage detection with composite hypothesis testing. In *2012 IEEE International Conference on Communications (ICC)*, pages 4883–4887. IEEE, 2012.
- Dapeng Liu, Youjian Zhao, Haowen Xu, Yongqian Sun, Dan Pei, Jiao Luo, Xiaowei Jing, and Mei Feng. Opprentice: Towards practical and automatic anomaly detection through machine learning. In *Proceedings of the 2015 Internet Measurement Conference*, pages 211–224, 2015.
- Yujing Liu, Xiapu Luo, Rocky KC Chang, and Jinshu Su. Characterizing inter-domain rerouting by betweenness centrality after disruptive events. *IEEE Journal on Selected areas in communications*, 31(6):1147–1157, 2013.
- Hans Peter Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of research and development*, 1(4):309–317, 1957.

- Yu Ma, Mugen Peng, Wenqian Xue, and Xiaodong Ji. A dynamic affinity propagation clustering algorithm for cell outage detection in self-healing networks. In *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 2266–2270. IEEE, 2013.
- Joseph Mathenge. Major network outages of 2021). <https://web.archive.org/web/20211021175645/https://www.bmc.com/blogs/network-outages/>, 2021. Accessed: 2021-12-22.
- Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. Openflow: enabling innovation in campus networks. *ACM SIGCOMM computer communication review*, 38(2):69–74, 2008.
- Dandan Miao, Xiaowei Qin, and Weidong Wang. Anomalous cell detection with kernel density-based local outlier factor. *China Communications*, 12(9):64–75, 2015.
- Behzad Mirkhanzadeh, Ali Shakeri, Chencheng Shao, Miguel Razo, Marco Tacca, Gabriele Maria Galimberti, Giovanni Martinelli, Marco Cardani, and Andrea Fumagalli. An sdn-enabled multi-layer protection and restoration mechanism. *Optical Switching and Networking*, 30:23–32, 2018. ISSN 1573-4277. doi: <https://doi.org/10.1016/j.osn.2018.05.005>. URL <https://www.sciencedirect.com/science/article/pii/S157342771730228X>.
- Thomas M Moerland, Joost Broekens, and Catholijn M Jonker. Model-based reinforcement learning: A survey. *arXiv preprint arXiv:2006.16712*, 2020.
- Jessica Moysen and Lorenza Giupponi. From 4g to 5g: Self-organized network management meets machine learning. *Computer Communications*, 129: 248–268, 2018. ISSN 0140-3664. doi: <https://doi.org/10.1016/j.comcom.2018.07.015>. URL <https://www.sciencedirect.com/science/article/pii/S0140366418300380>.
- M Arthur Munson. A study on the importance of and time spent on different modeling steps. *ACM SIGKDD Explorations Newsletter*, 13(2):65–71, 2012.
- Uma Narayanan, Athira Unnikrishnan, Varghese Paul, and Shelbi Joseph. A survey on various supervised classification algorithms. In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pages 2118–2124. IEEE, 2017.
- Aye Myat Myat Paing. Analysis of availability model based on software aging in sdn controllers with rejuvenation. In *2020 IEEE Conference on Computer Applications (ICCA)*, pages 1–7, 2020. doi: 10.1109/ICCA49400.2020.9022818.
- Julio-Omar Palacio-Niño and Fernando Berzal. Evaluation metrics for unsupervised learning algorithms. *arXiv preprint arXiv:1905.05667*, 2019.
- Neena Pandey, Abhipsa Pal, et al. Impact of digital surge during covid-19 pandemic: A viewpoint on research and practice. *International journal of information management*, 55:102171, 2020.

- Rafael Pasquini and Rolf Stadler. Learning end-to-end application qos from open-flow switch statistics. In *2017 IEEE Conference on Network Softwarization (Net-Soft)*, pages 1–9. IEEE, 2017.
- Shahzad Qaiser and Ramsha Ali. Text mining: use of tf-idf to examine the relevance of words to documents. *International Journal of Computer Applications*, 181(1):25–29, 2018.
- Kun Qiu, Jin Zhao, Xin Wang, Xiaoming Fu, and Stefano Secci. Efficient recovery path computation for fast reroute in large-scale software-defined networks. *IEEE Journal on Selected Areas in Communications*, 37(8):1755–1768, 2019.
- Lin Quan, John Heidemann, and Yuri Pradkin. Detecting internet outages with precise active probing (extended). *USC/Information Sciences Institute, Tech. Rep*, 2012.
- Lin Quan, John Heidemann, and Yuri Pradkin. Trinocular: Understanding internet reliability through adaptive probing. *ACM SIGCOMM Computer Communication Review*, 43(4):255–266, 2013.
- Lin Quan, John Heidemann, and Yuri Pradkin. Visualizing sparse internet events: Network outages and route changes. *Computing*, 96(1):39–51, 2014.
- Walter Quattrociocchi, Guido Caldarelli, and Antonio Scala. Self-healing networks: redundancy and structure. *PloS one*, 9(2):e87986, 2014.
- Juan Ramiro and Khalid Hamied. *Self-organizing networks: self-planning, self-optimization and self-healing for GSM, UMTS and LTE*. John Wiley & Sons, 2011.
- Sebastian Raschka. *Python machine learning*. Packt publishing ltd, 2015.
- Samira Rezaei, Hamidreza Radmanesh, Payam Alavizadeh, Hamidreza Nikoofar, and Farshad Lahouti. Automatic fault detection and diagnosis in cellular networks using operations support systems data. In *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, pages 468–473. IEEE, 2016.
- Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20: 53–65, 1987.
- Sinan Saraçlı, Nurhan Doğan, and İsmet Doğan. Comparison of hierarchical cluster analysis methods by cophenetic correlation. *Journal of inequalities and Applications*, 2013(1):1–8, 2013.
- Dominik Schatzmann, Simon Leinen, Jochen Kögel, and Wolfgang Mühlbauer. Fact: Flow-based approach for connectivity tracking. In *International Conference on Passive and Active Network Measurement*, pages 214–223. Springer, 2011.
- Aaron Schulman and Neil Spring. Pingin’ in the rain. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 19–28, 2011.

- M Zubair Shafiq, Lusheng Ji, Alex X Liu, Jeffrey Pang, Shobha Venkataraman, and Jia Wang. Characterizing and optimizing cellular network performance during crowded events. *IEEE/ACM Transactions On Networking*, 24(3):1308–1321, 2016.
- Dalwinder Singh and Birmohan Singh. Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, 97:105524, 2020.
- Jessica Steinberger, Anna Sperotto, Harald Baier, and Aiko Pras. Collaborative attack mitigation and response: a survey. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 910–913. IEEE, 2015.
- Péter Szilágyi and Szabolcs Nováczki. An automatic detection and diagnosis framework for mobile communication systems. *IEEE transactions on Network and Service Management*, 9(2):184–197, 2012.
- Sayali Sunil Tandel, Abhishek Jamadar, and Siddharth Dudugu. A survey on text mining techniques. In *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, pages 1022–1026. IEEE, 2019.
- Soon Tee Teoh, Supranamaya Ranjan, Antonio Nucci, and Chen-Nee Chuah. Bgp eye: a new visualization tool for real-time detection and analysis of bgp anomalies. In *Proceedings of the 3rd international workshop on Visualization for computer security*, pages 81–90, 2006.
- Brian Tierney, Joe Metzger, Jeff Boote, Eric Boyd, Aaron Brown, Rich Carlson, Matt Zekauskas, Jason Zurawski, Martin Swany, and Maxim Grigoriev. perf-sonar: Instantiating a global network measurement framework. *SOSP Wksp. Real Overlays and Distrib. Sys*, 2009.
- Ha Thi Thu Trang and Pham Thi Thanh Hong. Measuring the impact of infrastructure quality on firm performance: a review of literature, metrics, and evidence. In *International Conference on Emerging Challenges: Business Transformation and Circular Economy (ICECH 2021)*, pages 211–220. Atlantis Press, 2021.
- Rüdiger Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, volume 1, pages 29–39. Springer-Verlag London, UK, 2000.
- Tzu-Tsung Wong. Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. *Pattern Recognition*, 48(9):2839–2846, 2015.
- Jian Wu, Ying Zhang, Z Morley Mao, and Kang G Shin. Internet routing resilience to failures: analysis and implications. In *Proceedings of the 2007 ACM CoNEXT conference*, pages 1–12, 2007.
- Yang Xiang, Zhiliang Wang, Xia Yin, and Jianping Wu. Argus: An accurate and agile system to detecting ip prefix hijacking. In *2011 19th IEEE International Conference on Network Protocols*, pages 43–48. IEEE, 2011.
- Junfeng Xie, F. Richard Yu, Tao Huang, Renchao Xie, Jiang Liu, Chenmeng Wang, and Yunjie Liu. A survey of machine learning techniques applied to software

- defined networking (sdn): Research issues and challenges. *IEEE Communications Surveys Tutorials*, 21(1):393–430, 2019. doi: 10.1109/COMST.2018.2866942.
- Xin-She Yang. Chapter 14 - multi-objective optimization. In Xin-She Yang, editor, *Nature-Inspired Optimization Algorithms*, pages 197–211. Elsevier, Oxford, 2014. ISBN 978-0-12-416743-8. doi: <https://doi.org/10.1016/B978-0-12-416743-8.00014-2>. URL <https://www.sciencedirect.com/science/article/pii/B9780124167438000142>.
- Peng Yu, Fanqin Zhou, Tao Zhang, Wenjing Li, Lei Feng, and Xuesong Qiu. Self-organized cell outage detection architecture and approach for 5g h-cran. *Wireless Communications and Mobile Computing*, 2018, 2018.
- Tao Zhang, Lei Feng, Peng Yu, Shaoyong Guo, Wenjing Li, and Xuesong Qiu. A handover statistics based approach for cell outage detection in self-organized heterogeneous networks. In *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pages 628–631. IEEE, 2017.
- Tao Zhang, Kun Zhu, and Dusit Niyato. A generative adversarial learning-based approach for cell outage detection in self-organizing cellular networks. *IEEE Wireless Communications Letters*, 9(2):171–174, 2019.
- Ronggang Zhou, Xiaorui Wang, Yuhan Shi, Renqian Zhang, Leyuan Zhang, and Haiyan Guo. Measuring e-service quality and its importance to customer satisfaction and loyalty: an empirical study in a telecom setting. *Electronic Commerce Research*, 19(3):477–499, 2019.
- Zhenyu Zhou, Theophilus Benson, Marco Canini, and Balakrishnan Chandrasekaran. Delorean: Using time travel to avoid bugs and failures in sdn applications. In *Proceedings of the Symposium on SDN Research*, pages 199–200, 2017.

Appendices

Appendix A

Workplan Timelines

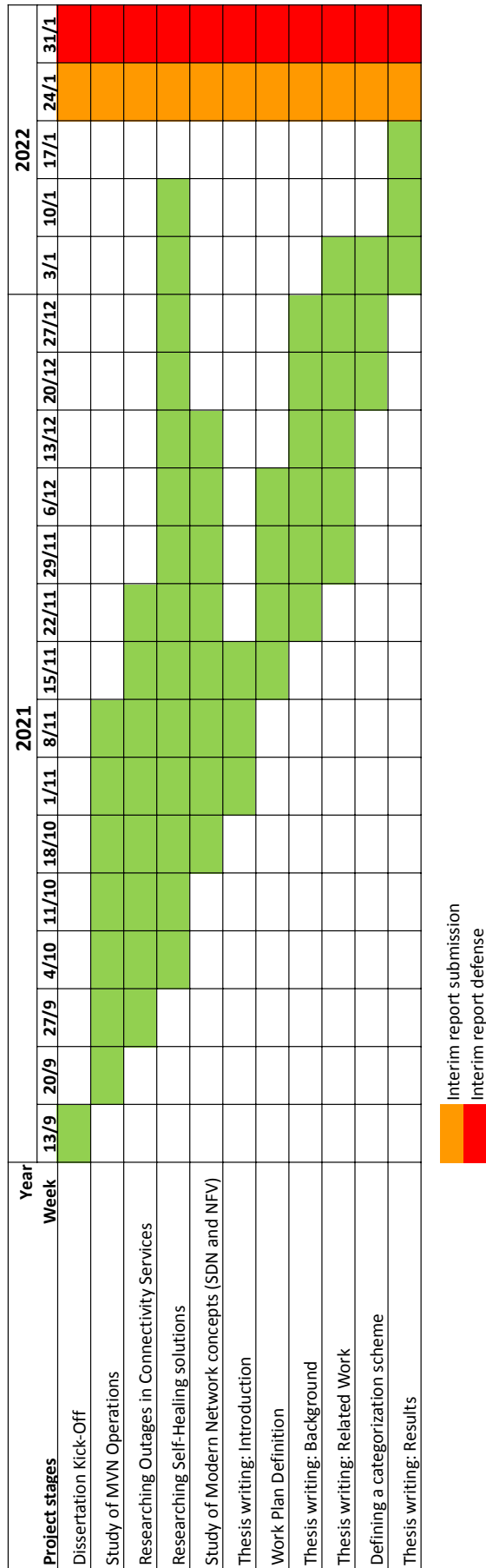


Figure A.1: Timeline for semester 1

		2022																			
		Year	7/2	14/2	28/2	14/3	28/3	11/4	25/4	9/5	23/5	6/6	20/6	4/7	18/7	1/8	15/8	29/8	5/9	7/9	
Research stages	Project macrotasks	Sprint																			
	Outage categorization	Analysis and categorization of outages @ Truphone																			
Self-Healing	Selection of outage type to handle																				
	Data exploration and infrastructure and protocol research																				
	Development of the outage detection mechanism																				
	Development of the outage self-healing mechanism																				
	Integration of the detection and healing mechanisms																				
Documentation	Solution integration/deployment planning																				
	Validation, Testing and Assessment of the solution																				
	Thesis writing, review and iterative enhancement																				
Milestones	Beginning of Fellowship @ Truphone																				
	End of Fellowship @ Truphone																				
	First prototype-detection																				
	Second prototype - detection and self-healing																				
	Thesis submission																				
	Thesis defense																				

Figure A.2: Timeline for semester 2

Appendix B

Key Truphone elements in this dissertation

Role	Contribution and Relevance
Head of R&D	This person's intimate knowledge of the company's business and architecture proved invaluable to guide the thesis in a direction that would create the greatest impact possible and to arrange meetings with key company elements that were crucial in the progress of the thesis.
R&D Software Engineers	They were instrumental in providing technical support and knowledge throughout all the stages of the thesis, and to more closely follow developments in the project.
Principal Engineer	As the Principal Engineer of Truphone, this person's technical knowhow and insights were extremely valuable to get a better grasp of the technologies involved in the different outages that were analyzed and to better understand which outages fell in the scope of the project and which did not.
Front Office Team	The Front Office team are usually the first responders in the event of an outage, and are thus very knowledgeable about Truphone's outage history, detection and handling methods, and perhaps more importantly where a self-healing system could be most impactful.
Network Service Manager	This person is deeply involved in the process of evaluating an outage's impact for clients and communicating outages internally and externally. By being so involved in this process, he possesses many insights into what are the necessities of Truphone and how to search and understand reports to find the best candidates for self-healing.
Network Engineers	Naturally, a crucial part of the process is understanding the affected systems and every underlying technology that may be relevant for the outage detection and healing processes. To that end, a constant back-and-forth dialogue with the engineers responsible for the development, maintenance and monitoring of the network's technologies.

Table B.1: Key Truphone members for the development of the thesis.

Appendix C

Incident Room event flow

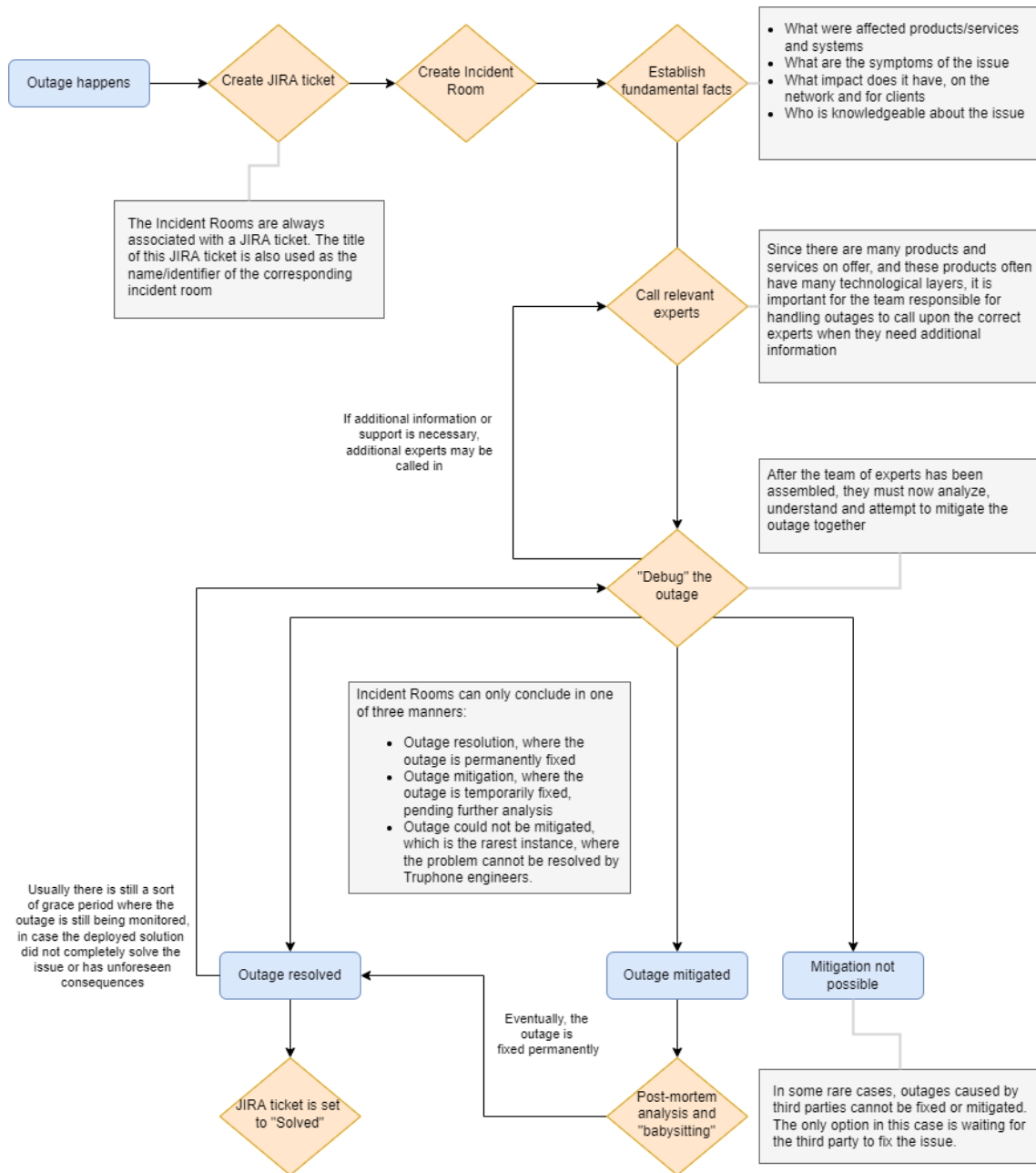


Figure C.1: Incident Room event flow

Appendix D

Simulator Data Examples

calling_address	called_address	ftn	start_date_utc	pool_country	country_code	pool
5636682468	4439157512	2829217091	469	A	A	A
1564054016	4341148890	1963877304	1445	A	A	A
5394723444	3822707293	1264553974	1783	A	A	A
5199592327	2022106545	3879510560	919	A	A	A
4474488948	7266687760	2498197095	2065	A	A	A
3962374119	1334554321	7105480200	603754	C	C	C
8564642894	3487959361	8938839115	602410	C	C	C
1959856444	4640645755	8660261583	603333	C	C	C
6078255199	9062167914	7351432971	602378	C	C	C
1715504743	9062167914	7300539375	601681	C	C	C

Table D.1: Simulated CAP CDR data set

calling_address	called_address	start_date_utc	start_event	call_id
5636682468	2829217091	478	3	1
1564054016	1963877304	1445	3	2
5394723444	1264553974	1785	3	3
5199592327	3879510560	927	3	4
4474488948	2498197095	2072	3	5
7304826282	7300539375	603194	3	137295
3962374119	7105480200	603759	3	137296
8564642894	8938839115	602411	3	137297
1959856444	8660261583	603351	3	137298
6078255199	7351432971	602381	3	137299

Table D.2: Simulated SIP CDR data set

Appendix E

Detection Architecture

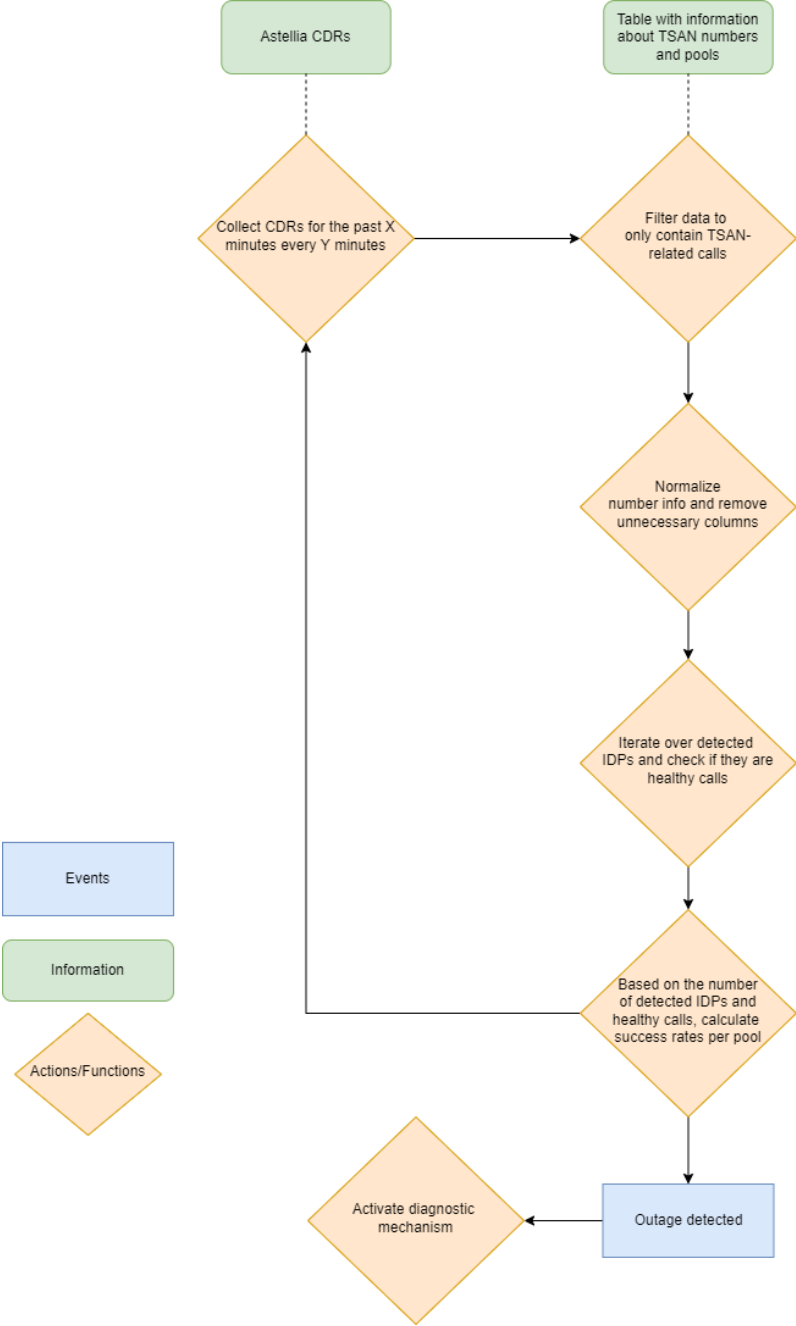


Figure E.1: Detection mechanism architecture

Appendix F

Healing Architecture

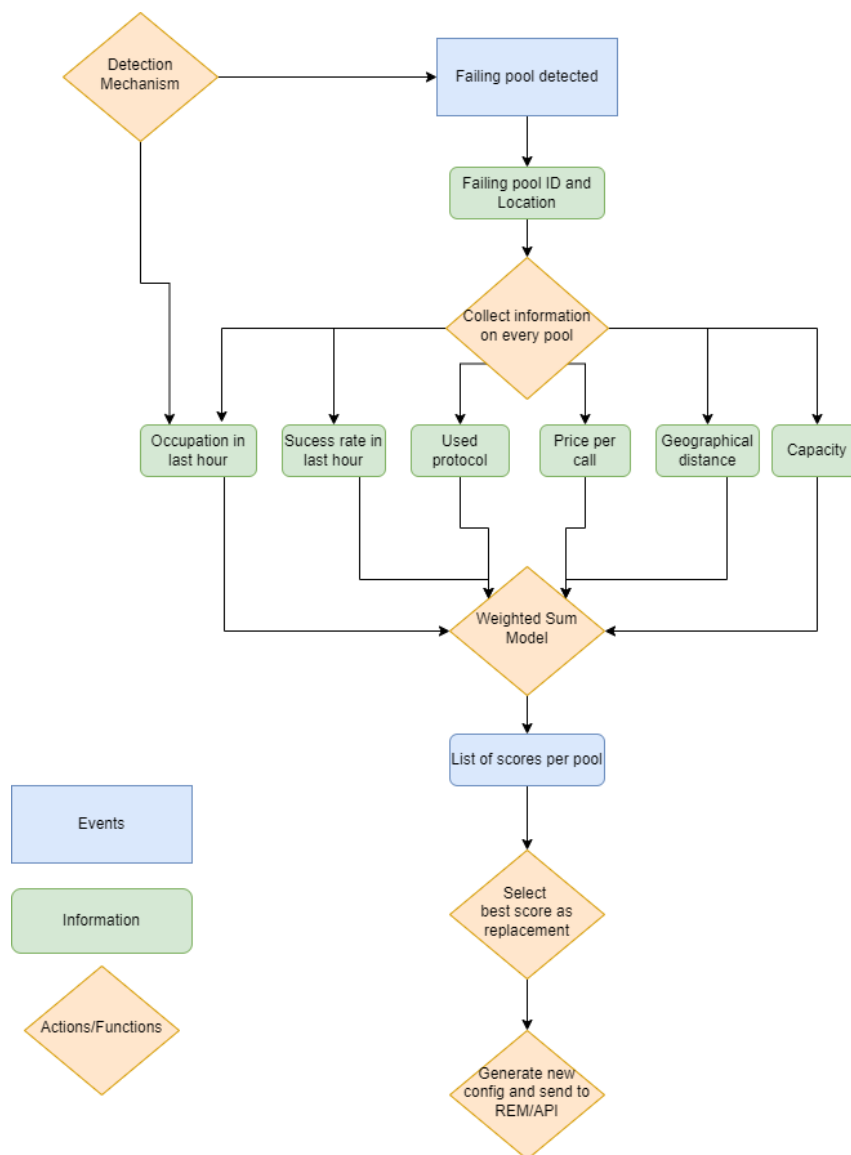


Figure F.1: Healing mechanism architecture

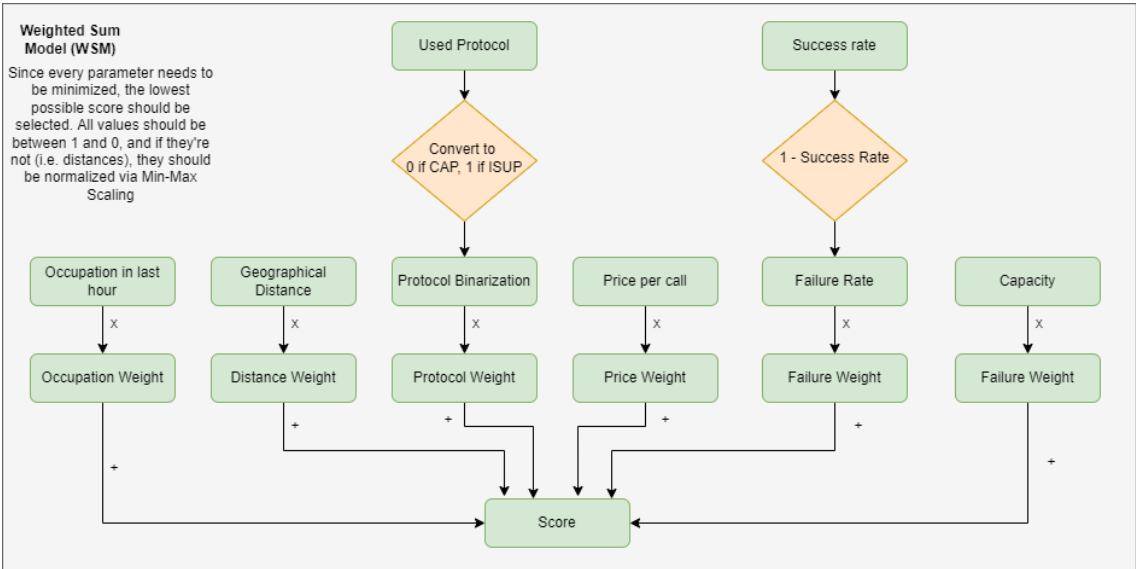


Figure F.2: Weighted Sum Model description