SPADAR : Situation-aware and proactive analytics for dynamic adaptation in real time

Mohammed Al Saleh

THESE DE DOCTORAT

NNT : 2022UPASG060

SPADAR: Situation-aware and proactive analytics for dynamic adaptation in real time

SPADAR : Analyse proactive et sensible au contexte pour l'adaptation dynamique en temps réel

Thèse de doctorat de l'université Paris-Saclay et de l'université Libanaise

École doctorale n° 580 sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Informatique
Graduate School : Informatique et sciences du numérique
Référent : Université de Versailles Saint-Quentin-en-Yvelines

Thèse préparée dans les unités de recherche DAVID (Université Paris-Saclay, UVSQ), et dans l'unité de recherche LAEC-CNRS (Université Libanaise), sous la direction de **Béatrice Finance**, Maître de Conference (HDR), et la codirection de **Ali Jaber**, Professeur, et le co-encadrement de **Yehia TAHER**, Maître de Conférence.

Thèse soutenue à Versailles, le 19 juillet 2022, par

Mohammed AL SALEH

Composition du jury

Walid Gaaloul Professeur, Université Telecom SudParis	Président du jury
Mahmoud Barhamgi Maître de Conference (HDR), Université Lyon 1	Rapporteur & Examinateur
Salima Benbernou Professeur, Université de Paris	Rapporteur & Examinatrice
Bilal Nsouli Directeur de recherche (Director of Lebanese Atomic Energy Commission), CNRS-LB	Examinateur
Yehia Taher Professeur, Maître de Conference, Université Versailles Paris-Saclay	Examinateur
Emmanuel Waller Maître de Conférences, LRI, Paris-Saclay	Examinateur
Béatrice Finance Maître de Conference (HDR), Université Versailles Paris-Saclay	Directrice de thèse

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre: SPADAR : Analyse proactive et sensible au contexte pour l'adaptation dynamique en temps réel
Mots clés: Algorithmes d'apprentissage automatique, modèle prédictif, regroupement de séries temporelles, classement des séries temporelles, débit de dose gamma, réseau d'alerte précoce aux rayonnements

Résumé: Bien que le niveau de rayonnement soit une préoccupation sérieuse qui nécessite une surveillance continue, de nombreux systèmes existants sont conçus pour effectuer cette tâche. Radiation Early Warning System (REWS) est l'un de ces systèmes qui surveille le niveau de rayonnement gamma dans l'air. Un tel système nécessite une intervention manuelle élevée, dépend totalement de l'analyse d'experts et présente des lacunes qui peuvent parfois être risquées. Dans cette thèse, l'approche RIMI (Refining Incoming Monitored Incidents) sera introduite, qui vise à améliorer ce système pour gagner en autonome tout en laissant la décision finale aux experts. Une nouvelle méthode est présentée qui aidera à changer ce système pour devenir plus intelligent tout en apprenant des incidents passés de chaque système spécifique.

Title: SPADAR: Situation-aware and Proactive Analytics for Dynamic Adaptation in Real time
Keywords: Machine learning algorithms, predictive Model, time series clustering, time series classification, gamma dose rate, radiation early warning network.

Abstract: Although radiation level is a serious concern that requires continuous monitoring, many existing systems are designed to perform this task. Radiation Early Warning System (REWS) is one of these systems which monitors the gamma radiation level in the air. Such a system requires high manual intervention, depends totally on experts' analysis, and has some shortcomings that can be risky sometimes. In this thesis, the RIMI (Refining Incoming Monitored Incidents) approach will be introduced, which aims to improve this system while becoming more autonomous while keeping the final decision to the experts. A new method is presented which will help in changing this system to become more intelligent while learning from past incidents of each specific system.

Le niveau de rayonnement environnemental est une préoccupation extrêmement sérieuse en raison de son impact catastrophique aussi bien sur les êtres vivants et que sur l'environnement. L'analyse des données de rayonnement avec une grande précision constitue un problème très difficile en raison de plusieurs obstacles. Par exemple, les fausses alertes positives générées par le Radiation Early Warning System représentent un problème courant dans l'analyse du niveau de rayonnement. Plus précisément, considérons le cas où l'analyse historique de mesures indique une augmentation du niveau de rayonnement qui dépasse un seuil prédéfini produisant une alerte de menace. Cependant, il peut s'agir d'une fausse alerte due à une corrélation insignifiante (elle peut être connue ou inconnue) entre des paramètres. Par exemple, si le niveau de rayonnement augmente en raison des conditions météorologiques, en particulier un climat pluvieux, l'alerte générée par le système est insignifiante. Ces types d'alertes sont appelées, dans le cadre de nos recherches, des « alertes innocentes » qu'il faut prévenir. Par conséquent, toute alerte de rayonnement doit être vérifiée avec une grande précision. Actuellement, le processus de vérification est principalement manuel, qui dépend totalement d'analyse d'experts et présente des lacunes qui peuvent parfois être risquées. Une solution automatisée de vérification des alertes représente une exigence essentielle pour améliorer l'efficacité du système. À l'heure actuelle, il n'existe aucune solution permettant de répondre à cette exigence.

L'avènement des technologies de l'information a permis de développer des systèmes automatisés qui réduisent l'effort humain et augmentent l'efficacité dans l'exécution des tâches dans divers domaines. Ces dernières années, les agences de surveillance des rayonnements ont tiré parti de la puissance de l'informatique pour surveiller le niveau de rayonnement. Certaines agences nationales ont commencé à adopter des technologies modernes, en particulier des capteurs pour surveiller et collecter des données de rayonnement à partir de différents emplacements géographiques. Ils collectent, stockent, et analyses ces données dans le but de détecter de niveaux de rayonnement élevés posant des menaces. Cependant, l'écosystème technologique actuel utilisé par les systèmes de surveillance du niveau de rayonnement (tel que le Radiation Early Warning System) a une capacité limitée qui doit être abordée pour développer un système entièrement fonctionnel et hautement efficace.

Notre travail de recherche a révélé plusieurs limitations des Radiation Early Warning Systems existants. Ci-dessous la liste des principales limitations qui doivent être traitées pour développer un écosystème hautement efficace :

- Absence d'infrastructure pour développer et déployer un écosystème à grande échelle piloté par les objets connectés.
- Absence de système prédictif : le système actuel de surveillance du niveau de rayonnement est réactif. En effet, ce système produit un rapport ou crée une alerte lorsque le niveau de rayonnement dépasse un certain seuil ; Il ne permet pas la prévention à l'avance d'un tel événement.
- Absence d'outil de vérification autonome : le calcul du niveau de rayonnement est une agrégation de différentes valeurs de différents paramètres, notamment la température, l'état météorologique et autres. Les valeurs de ces paramètres ont une influence importante pour déduire si le niveau de rayonnement est innocent ou pas. À notre connaissance, dans certains cas, le niveau de rayonnement peut augmenter en raison de la pluie, qui revient à des valeurs normales après un certain temps. Actuellement, la vérification des alertes innocentes se fait manuellement.

Dans cette thèse, nous introduisons l'approche RIMI (Refining Incoming Monitored Incidents) qui vise à améliorer le Radiation Early Warning System pour gagner en autonomie tout en laissant la décision finale aux experts. RIMI s'appuie sur l'apprentissage automatique pour une vérification automatique et en temps réel des alertes générées par le système. Une telle vérification affectera un incident, avec une grande précision, à une classe prédéfinie des incidents parmi un ensemble des classes formées sur la base de l'analyse de données historiques.

RIMI permettra non seulement de de prédire la nature de l'incident, mais aussi de déterminer la cause qui pourrait être à l'origine de celui-ci. Il s'agit également d'un système évolutif qui continue, à fur et à mesure, à apprendre des incidents et à adapter les classes d'incidents en fonction de l'évolution des situations.

Finalement, RIMI permettra de répondre aux questions telles que pourquoi la pollution par rayonnement s'est produite ou pourquoi le niveau de rayonnement a augmenté. RIMI é permettra d'identifier de corréla-

tions insignifiantes et réduira donc les coûts en termes d'effort manuel et de temps requis pour traiter les fausses alertes. Cela se traduira par la prédiction de la cause du niveau de rayonnement après avoir analysé les cas et scénarios précédents.

Contents

List of Figures

List of Tables

1 - Introduction

Contents

1.1 . Context & Motivation

The radiation level is one of the most critical hazards that must be taken care of due to its catastrophic and persistent consequences on the environment, humans, and other living things. This level is a critical concern due to its detrimental impact on living beings and the environment. Although there are different types of radiation (e.g., alpha radiation, gamma radiation) stemming from various radioactive materials and natural resources, the higher level of this radiation, specifically gamma radiation, causes severe damage to human health. Therefore, controlling radiation levels is critically important. In order to control radiation levels, monitoring radiation sources and analyzing radiation data are indispensable tasks.

Radioactive incidents and disasters such as Chernobyl [1], Fukushima [2], and the most recent one at the Russian nuclear missile test site [3], raised a serious concern since they led to the increase of radiation pollution levels in the atmosphere. This can be detrimental to the global environment and by extension to human civilization. Such events have given rise to the need for continuous monitoring of the radiation level in the environment. Thus, this certainly gives rise to a critical question, *what is the most suitable approach to monitor the radiation level*?

Monitoring the radiation level in the environment is very critical. Long term or acute exposure to a high gamma dose rate can have many hazardous consequences on humans as well as the ecosystem [4]. A serious event that occurs and causes an abrupt increase in the gamma dose rate can be what happened in the Chernobyl accident where the biggest short term leak of radioactive materials was ever recorded in history [4]. Such an event has to be intercepted at the earliest point possible to take the proper measures and precautions and notify the concerned authorities to seek to minimize the effects of such a hazardous situation.

Since the radiation can be transmitted through the wind, monitoring the radioactivity within widespread geographical locations is important to prevent any unwanted exposure. The continuous monitoring would greatly help in taking a proactive measure that would eventually raise an alert upon an occurrence of the incidence. Therefore, many countries around the world raised the idea of developing several techniques for monitoring the radiation level in the environment to detect any abnormal release or discharge. These countries developed national environmental radiation monitoring programs to establish a radiation baseline level and determine the trend of radiation level. Air monitoring was one of the main scopes of these programs.

Over the last few decades, several approaches have been developed to control and monitor the radiation level in the environment. Among them is the Radiation Early Warning System (REWS) [5] which is a widely used network system that exists in many countries around the world [6] [7]. It can help in taking action in advance in abnormal cases. The REWS has an alert system to notify possible high gamma dose radiation. It enables the users to set different parameters used as the

system's reference values. Using these systems, the gamma dose rate is monitored in real-time and whenever an event occurs, where the gamma dose rate goes above the accepted peak value provided by experts, an alarm is triggered. Thus, a team of experts and personnel have to unite to investigate the reasons behind this rise. After several verifications of the use case, necessary procedures must be taken if an emergency is detected.

Using the Radiation Early Warning System (REWS), experts monitor the gamma dose rate to intercept any high rate incidents to detain a possible real threat. However, as the incident could be *real*, *innocent*, or false, they have to manually analyze multiple data sources before deciding on the reason and acting accordingly. This will introduce a framework for realizing an Intelligent REWS that will be able to autonomously recognize the causes of such incidents in real time with as little human intervention as possible.

The REWS is composed of many radiation detection sensors (also called probes) disseminated in a specific region that monitors the gamma radiation level. This system reacts as soon as possible to anomalies by raising an alert. Typically, the alerts are determined by predefined peak values that are essentially chosen based on observations (i.e., experience) [8]. It is worth noting that there are different peak values at different locations since the peak value depends strictly on the normal reading of the radiation level (known as background level) which is, in turn, not fixed due to many factors such as the altitude. Once an alert is raised, it needs to be checked by an expert. Indeed, the expert needs to analyze the potential causes of the incident as some alerts refer to an authentic threat of high radiation level and others denote the rise of radiation level that has no hazardous impact on the environment or living beings. In order to do so, the expert will consult additional information such as the weather broadcast and the quality factors (also called quality bits) of the probe.

For instance, the alert is false when the quality bits of the probe indicates that there is a defect in the probe [9], meaning that we cannot trust the collected gamma dose rate value. The alert is innocent when external factors have occurred such as rain, wind, lightning, etc... These external factors are the more challenging to analyze, but they represent more than 90% of the alarms. Finally, if the alert is real and an emergency action needs to be taken by the authority immediately.

The ecosystem of the used traditional radiation early warning system helped us highlight its shortcomings. The main purpose of this dissertation is to distinguish between the system's three types of alarm: innocent, false, and real knowing that the current system only provides alarms based on the reached threshold. Furthermore, in order to be able to classify the three different types of alarms, we must be able to determine the cause of the high gamma dose rate.

The overall objective of the research is to enhance the quality of the radiation monitoring program globally. This will be done by developing the technical capabilities for monitoring the radiation levels. In this dissertation, we focus more

on technical challenges and limitations of radiation safety. Improving the radiation early warning system to become fully automated significantly impacts the radiation monitoring field. It will improve the monitoring process by limiting the time and the effort needed upon alerts. It will also notify the experts to take specific procedures upon real threats as soon as possible. This will shorten the time needed to distinguish between real, innocent, and false alerts of high radiation levels.

1.2 . Challenges

After going through the ecosystem and the context behind the radiation early warning system, we can identify multiple existing problems with different aspects. The REWS has some powerful features that enable to detect and alert incidents. However, it has some important shortcomings. The first problem is that there is no generic solution. Every country has its own parameters to monitor. So one possible model may work in a certain country but will seem unfit in another.

Another problem that should be mentioned is the background value which the experts rely on to carry out analysis. The background in a certain region may not be fixed, it may evolve over time due to the aging of the probes or climate change. The difference in the background values between regions may also be due to the location of the probes. A country which passes through a historical radioactive incident may have a higher background level than other countries.

The gamma dose radiation is very sensitive to the environment. When it snows, the soil and the surroundings get covered by the snow. This leads to an important decrease in the gamma dose rate. Environmental change also can be a problem. Since the weather, the season, and global warming are all changing through the years, a new problem has risen. A change in the temperature or the pressure also leads to changes in the background level. The expert said that they use the average of the past two weeks in order to estimate the background. The evolving of the background creates a confidence problem in the data and the estimation.

Moreover, the radiation early warning system requires significant manual interventions and experts' opinions to perform causal analysis which is required for investigating an alert. Experts spend time and effort to identify the genuine cause of an alert which can be real, innocent, or false. Innocent alerts may be caused by external factors such as weather. For example, *rain* increases the radiation level which would reach the peak value; this might lead to a threat that is similar to the real threat. In general, the peak value of radiation engendered by rain needs several hours to roll back to the normal value which mimics a real incident. Therefore, this becomes highly challenging for experts to know the genuine cause of the incidence in real time since the process is human driven. Alerts generated because of such cause, require a long analysis procedure before detecting the main cause behind the alarm. This requires weather data collection and analysis, investigating the surroundings of the probe for any intruders, and monitoring the radiation level

for several hours after the alarm to make sure that it will return to normal values. This can be a true risky situation in case of a real thread is happening.

Currently, this system depends on experts' experiences in order to define the type of alarm. This requires many procedures upon having an alarm starting from checking many parameters (internal or external) up to visiting the affected site to investigate the alarm. It is worth noting that the causes of the radiation incidents could vary from real to innocent threats, yet critical factors such as time, identifying the cause, and analyzing different data sources are vital for the investigation process. Currently, the investigation process is purely manual due to the lack of automation support in REWS. Also, REWS are built on conventional technologies that are static. Hence, the capabilities of REWS are limited.

Performing deeper investigations, we found several limitations for the existing radiation early warning system. Such limitations should be addressed to develop a highly efficient ecosystem. The most important salient limitations behind the current system are:

- Lack of infrastructure for developing and deploying a full-scale IOT driven ecosystem.
- Lack of proactive/predictive system: The current radiation level monitoring system is reactive, that is, this system produces a report or creates an alert when the radiation level exceeds a certain peak value. However, predicting the cause behind such an occurrence is not possible with the existing monitoring system.
- Lack of Autonomic Verification tool: Analyzing radiation level is an aggregation of different values of different parameters including temperature, weather status, and others. The values of these parameters have a huge influence on deciding if the increase in the radiation level is real or not. Currently, the verification of the raised alarm is done manually.

1.3 . Objectives and Contributions

The advent of information technologies (IT) has enabled to development of automated systems that reduce human effort and increase efficiency in doing tasks within various areas and domains. In recent years, radiation monitoring agencies are leveraging the power of IT to monitor radiation levels. Some national agencies have started using modern technologies, in particular, sensors to monitor and collect pollution data from different geographical locations within the country. They store the radiation data and study them to detect a higher level of radiation which poses huge threats. However, the current technological ecosystem used for the radiation early warning system has a limited capacity that must be addressed to develop a fully functional and highly efficient system that ensures safety to all living organisms from high radiation levels.

One of the main objectives behind this research is to take a deeper look at the radiation early warning system and focus on its shortcomings. The main problem behind the REWS system is that it requires a lot of manual interventions and experts' opinions during the analysis process of raised alerts. This can be sorted out by an automatic system that does not exist.

Several approaches were investigated in order to discover the possible solutions that can target the shortcomings behind the radiation early warning system. To the best of our knowledge, there is no existing previous approach that can target the whole issues in the mentioned system at the same time. Thus, the shortcomings were targeted separately in order to investigate the previous contributions to the existing issues at each stage.

Accordingly, by exploiting the power of information technology and the power of data science, these issues can be sorted out. We strongly believe that this manual intervention can be significantly reduced by an automated system that elaborates the power of data science and AI techniques as this will help in developing an advanced intelligent system that can tackle the shortcomings. However, we plan to change this current human driven ecosystem into a fully machine driven automated system.

Our objective of this research is to develop an autonomous and intelligent system that would exploit the power of highly advanced autonomic technologies such as machine learning and artificial intelligence while preserving the effective and efficient features of the current REWS systems. Gaining intelligence is the key aspect of our approach. Therefore, our aim was to transform the static and semi-automatic REWS systems into dynamic, fully automatic, and intelligence driven. The proposed system would optimize the ability to analyze any alert generated by an event such as rain. In this dissertation, we will present and discuss the design of the high-level architecture of our solution.

Since the gamma dose rate might be affected by several causes, distinguishing the innocent gamma dose rate from the real one is not an easy task for the experts. Through this dissertation, a machine learning based framework entitled RIMI (Refining Incoming Monitored Incidents) is proposed to automate the recognition of the radiation incidents in order to decrease the time and efforts spent as well as to increase the efficiency and accuracy of the process. The proposed framework aims to automate the process of monitoring the gamma dose rate produced by the running REWS. The system has to intercept incoming incidents, refine them, and detect their cause with a high level of accuracy in real-time so that they can be properly dealt with respectively.

The convention on technologies, for instance, is very difficult to tackle the issues behind the radiation early warning system. These issues will be addressed in this dissertation. Thus, to build an automated system, RIMI framework will present different phases assuming that each phase will tackle some challenges. These phases will cover investigating historical data, analyzing the peaks and ex-

tracting their respective incidents, searching for similar incidents, and comparing the incidents during the live data analysis.

Our objectives are two-fold in this research. First, we will improve the radiation early warning system to become a fully automated alert verification system that will verify every alert through the RIMI proposed framework based on an analytical model (which we will be developed within the scope of this dissertation). The analytical model will discover the main cause behind the high radiation level. Essentially, the proposed framework will answer why questions; for instance, why radiation pollution occurred or why radiation levels increased. The proposed framework will enable the identification of such insignificant correlation and hence will reduce cost in terms of manual effort and time required to deal with false and innocent alarms.

Next, we will trigger artificial intelligence to reconstruct the running model dynamically. The objective of this reconstruction process is to avoid false and innocent alarms. The reconstruction process will heavily rely on the outcomes of historical data analysis and the system will rebuild the descriptive analytical model dynamically. The reconstruction system will be built on the "Autonomic Computing" paradigm. We will investigate existing technologies such as "evolutionary algorithm" and will choose the best fit technologies which will enable our system to perform reconstruction of the model automatically. This will result in predicting the radiation level cause after analyzing previous cases and scenarios.

Therefore, the proposed RIMI framework is divided into two phases: (1) building a predictive model and (2) online incident detection. To develop the first phase, historical past data will be used to learn about the behavior of the incidents, in terms of shape (scale) and duration (length). This directed us to contribute to the time series clustering field where clustering of unlabeled time series data with different scales and lengths was required. The learning process will result in several classes that are grouping similar incidents. The obtained classes will highlight the different causes behind an increase in the gamma dose rate.

However, through the second phase, the system will compare the received data stream with the previously analyzed data stream. Thus, this guided us to contribute to the time series classification field where the incoming online incidents, with different scales and lengths, should be classified as soon as possible in order to propose a possible cause. After identifying similar incidents, the possible causes should be verified with the expert for further validation.

What we are hoping to reach in this dissertation is to, as autonomously as possible, end up with the different groups containing the incidents, extracted from the historical data, where each group can be explained by a different event occurring in the background so that these grouping (classes) can be used for real-time analysis of the gamma dose rate data.

1.4 . Structure of the Dissertation

Figure 1.1 introduces the thesis structure. Specifically, the rest of the thesis is structured as follows:

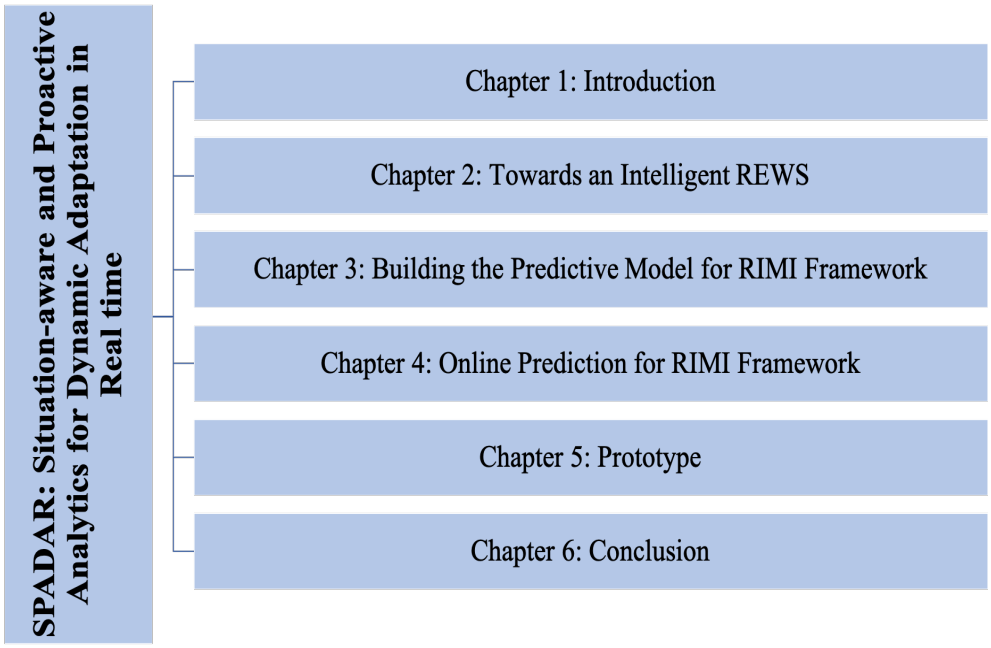

Figure 1.1: Dissertation structure organisation

- Chapter 2 reviews the traditional Radiation Early Warning System and its main components. It highlights the existing experts' roles in the current system and introduces the need for moving into an intelligent REWS through a proposed framework called RIMI.
- Chapter 3 introduces the first phase of the proposed RIMI framework. It highlights our contribution through combining several preprocessing methods and machine learning algorithms to build the predictive model of RIMI framework.
- Chapter 4 focuses on the second phase of the proposed RIMI framework. It highlights the faced issues behind using the existing classification methods. Moreover, it introduces our contribution in developing the online prediction phase.
- Chapter 5 presents the system architecture for the implementation of the whole end-to-end proposed framework.
- Chapter 6 provides the general conclusion behind this thesis. It also highlights some future work that can be done towards reaching a full automated intelligent Radiation Early Warning System.

1.5 . Scientific Contributions

Through this research work, several contributions have been introduced into the research field. Four articles have been published or submitted to different scientific journals or international conferences.

1. M. Al Saleh, B. Finance, Y. Taher, and A. Jaber. "Towards An Autonomous Radiation Early Warning System". In CEUR Workshop Proceedings. BDC-SIntell, Versailles, France, 2019.

2. M. Al Saleh, B. Finance, Y. Taher, R. Haque, A. Jaber and N. Bachir. "Introducing artificial intelligence to the radiation early warning system". Environmental Science and Pollution Research, Pages 14036-14045, 2021.

3. M. Al Saleh, B. Finance, Y. Taher, and A. Jaber. "Time Series Clustering of High Gamma Dose Rate Incidents", Accepted in 8th International conference on Time Series and Forecasting, Gran Canaria, Spain, 2022.

4. M. Al Saleh, B. Finance, Y. Taher, and A. Jaber. "Online Classification of High Gamma Dose Rate Incidents", Accepted in 8th International conference on Time Series and Forecasting, Gran Canaria, Spain, 2022.

2 - Towards an Intelligent Radiation Early Warning System (REWS)

Contents

One of the several effect of the radioactivity is the cancer and since radioactive materials can be transported though the wind or through the storms, it is a must to monitor the radioactivity in every populated region in order to make sure that there is no nuclear danger. The system used to monitor this type of environmental radioactivity is called Radiation Early Warning system (REWS) which is used in many countries around the world. The REWS is mainly used to monitor environmental Gamma Dose Radiation (GDR) that represent radiation in the area. The system sends alerts in case of a high GDR and after several verification of the use case, if an emergency is detected, necessary procedures must be taken.

Although the investigation process may lasts few hours before detecting the real cause behind the current situation, the current occurring incident could be either false, innocent or real incident depending on the factors behind it. Thus, the traditional REWS deals with many information needed by the experts in order to analyze the occurring situation. However, this raised the main objective behind this thesis which focuses on changing the current REWS into an intelligent one.

Knowing that the application domain behind the REWS is critical, it is good to mention here that the context of this thesis is a real one. I have been working at the Lebanese Atomic Energy Commission for the past 8 years where the Radiation Early Warning System is used for monitoring the environmental gamma radiation level in Lebanon. Being in contact with the experts responsible for the REWS, we discovered that the current system has many shortcomings that can be addressed in order to turn it to a full intelligent system. Moreover, we didn't face the problem of simulating data since the experts from Germany and Lebanon offered us the real data that was used in this thesis. Thus, this motivated us to introduce a real contribution that will be approved by the experts in order to address the real problems they are facing. The main purpose of this chapter is to introduce the context of the REWS and the motivations towards dealing with its shortcomings.

Throughout this chapter, we will first recall the ecosystem of the traditional REWS. Through the ecosystem, we will go through the main components of the REWS. Then, we will explain the role of the experts and the current intelligence followed by them in order to analyze the incoming incidents. Next, we will introduce the traditional REWS shortcomings where the problem statement and the objectives behind this thesis are explained. Later we will conclude with our proposed framework called RIMI (Refining the Incoming Monitored Incidents) that will help in converting the traditional REWS into a full intelligent one.

2.1 . Traditional REWS

In this section, we will introduce the main components that are describing the ecosystem behind the REWS. The system has a very simple architecture. It is composed of two main parts: the probe and the data center. The monitoring station also known as probe is playing the role of a sensor. In each region of

interest many probes are set in order to monitor and send the gamma dose rate. The data is sent by SMS or through an FTP connection to the data center. Both the probes and the data center have thresholds that trigger their altered state, it may or may not be the same. Noting that the probe is only able to monitor gamma dose rate and not the nature of the isotope causing this high gamma dose rate, so determining which isotope is causing this radiation requires the experts to go to that region and to use special devices. But the gamma dose rate might be affected by several different weather events such as rain, wind, pressure, temperature, snow or by some transported products. Sometimes, a defected element in the probe might send false readings. This leads to a very complex model where detecting if the high gamma dose rate is caused by an innocent event (like rain or a shock), or by erroneous reading (false alarms) or by an actual threat (real case). Note that the current system is unable to distinguish the false reading, from the innocent case scenarios from the real case (actual thread).

In this part we will talk about the components of the system along with its architecture and the process of communication between the two main parts.

2.1.1 . The Probe (Monitoring Station)

The probe, also known as the monitoring station by REWS experts, is equipped with different sensors. Note that each country decides the type of sensors they equip the probe with. Figure 2.1 shows the component of the probe used by the REWS. It contains the GammaTRACER XL2 GSM/SMS which provides a modem responsible of all the data transfer. The probe may also have a Rain Sensor which is a critical element for the important effect of the rain on the gamma dose rate. A Solar Power Panel is also used in order to provide power to the probe. It is connected to the GammaTRACER Extension Unit. The Extension Unit connects the GammaTRACER, the Rain Sensor, the Solar Power Panel, the alarm and the heater power (optional).

Now for the hardware of the probe, we will focus only on the most important elements. The probe contains two Geiger-Müller(GM) tubes. The first is low dose GM tube, it's a very sensitive sensor. While the second one is a high dose GM tube with a sensitivity 470 times smaller to the low dose. Concerning the internal additional sensors, we can also find temperature sensor, movement sensor, humidity sensor and a voltage monitor. Those sensors can be used to check the quality of the probe (i.e. if the voltage is below the required range than the probe is most probably not sending correct data. The same goes for the humidity in the probe, the movement and the temperature). As hardware extensions we can have an alarm unit, a GPS module (its usage is critical if the probe is moving or for security purposes in case the probe must be fixed and start moving) and an additional battery pack. Note that a RS485 module, along with a ShortLINK module, a SkyLINK module and a GSM module are all available on the probe and could be used for communication purposes.

Note that the main purpose of the probe is to monitor the gamma dose rate

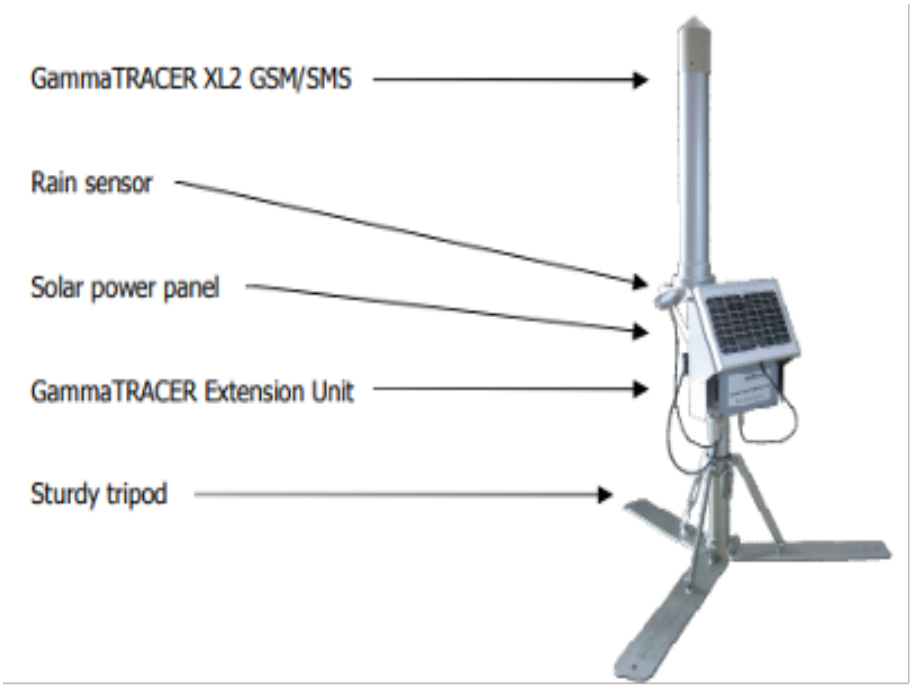

Figure 2.1: The components of the probe

in the environment as shown in Figure 2.2. It has a normal state in which the GDR is at a normal level. In some countries (like Lebanon where approximately 27 probes are deployed), at the normal state, the probe takes the average of the readings during one hour. It also sends the readings every six hours to the Network Monitoring Center. In other countries (like Germany where approximately 1000 probes are deployed) the data is transferred each minute.

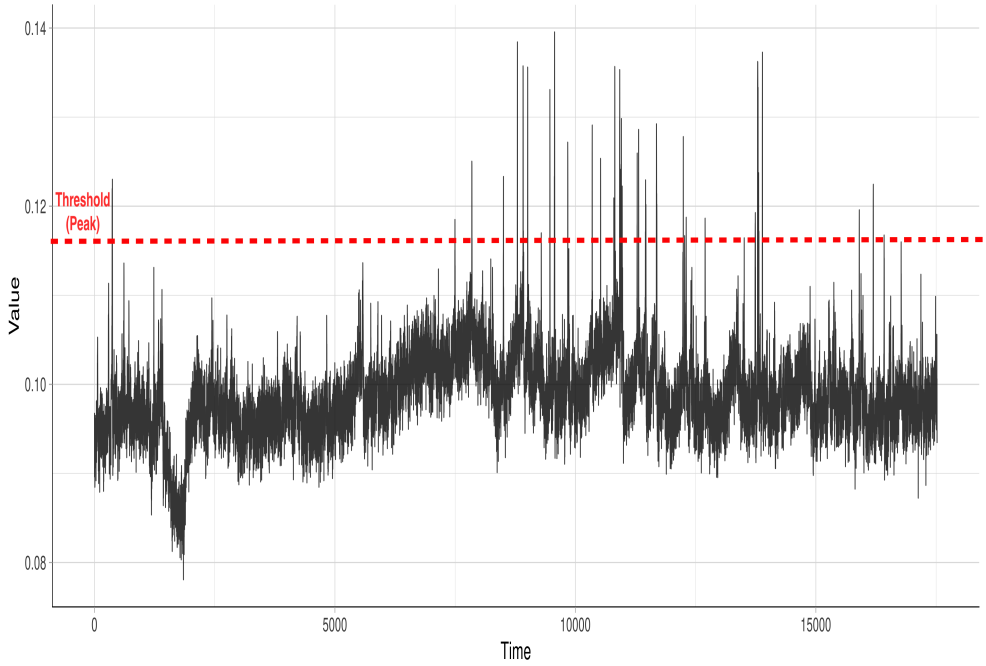

Figure 2.2: Gamma dose rate data of a certain probe in the year 2015

The intelligence in the probe is related basically on its own threshold (the threshold dilemma will be developed latter on in this section). In some countries (like Lebanon), the threshold at the data center may not be the same as the probe, it might be a little bit higher. If the GDR reach and surpass the threshold, then the probe will start automatically sending data more often. This allows more data so more details about how the gamma dose rate is evolving. Other countries (like Germany), the threshold at the probe is the same at the data center since the probe sends the gamma dose rate readings every minute to the data center.

Usually, the probe sends a message containing the gamma dose rate average within a specific interval of time depending on the alarm mode fixed by the experts in each country. In addition to the gamma dose rate, the probes send other sensors data (also known as Quality Bits) as they are equipped with internal sensors that can detect the defectiveness of any of the system components. These sensors data are stored in the historical REWS database for later analysis. The sent data contains the Gamma Dose rate along with 29 quality bits that shows the state of the probe. They are used to see if they can trust the data and to verify the quality of the probe and the data. For example, we have the quality bits that represent the state of the probe to check if a problem is detected in the probe or not. For instance, the humidity within the device must be below 30%, so if the detected humidity is above that value then the humidity bit will be equal to 1 otherwise it will be 0. It also points out an important part that concerns the internal and external factors that can affect the reading.

2.1.2 . The Data Center

The Data Center, also known as the Network Monitoring Center by REWS experts, plays the role of the data receiver, processor and database.

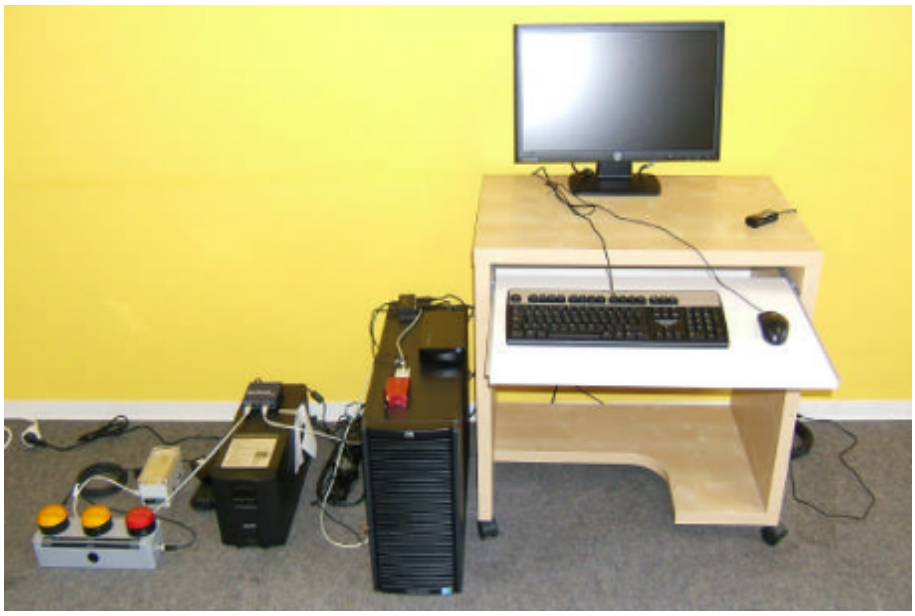

Figure 2.3: The components of the Data Center

Figure 2.3 shows the component of the data center. We can easily identify the server that is connected to the GSM modem. It allows the reception of the SMSs sent by the Probes forming the network. A UPS that last for an hour in case of a power failure and an External Backup Disk are used as a backup. A GPS clock is added in order to provide a correct system time and the synchronization of the network time. A network Alarm Box is connected to the server which will turn on to indicate that high gamma dose rate incidents are being received from the probe.

The Data Center stores the incoming gamma dose rate readings sent by the probe along with the quality bits in historical databases. Since the main role of the data center is to collect the data, it offers some beneficial screens to the experts to check the incoming incidents and classify them as real or unreal incidents. However, we should mention that other data types and information used by the experts during the analysis process are not stored in the historical databases including the incidents labeling after discovering the real cause behind them.

2.1.3 . Communication Process

The connection between the probes and the datas center could be either through M2M (Machine to machine) or FTP (File Transfer Protocol). These types of connections are used to allow the security of the transferred data. The data acquisition is done on a regular basis through secure channels between the

probes and the data center. Figure 2.4 shows the previously mentioned interaction between the probe and the data center. It also shows the type of the data sent.

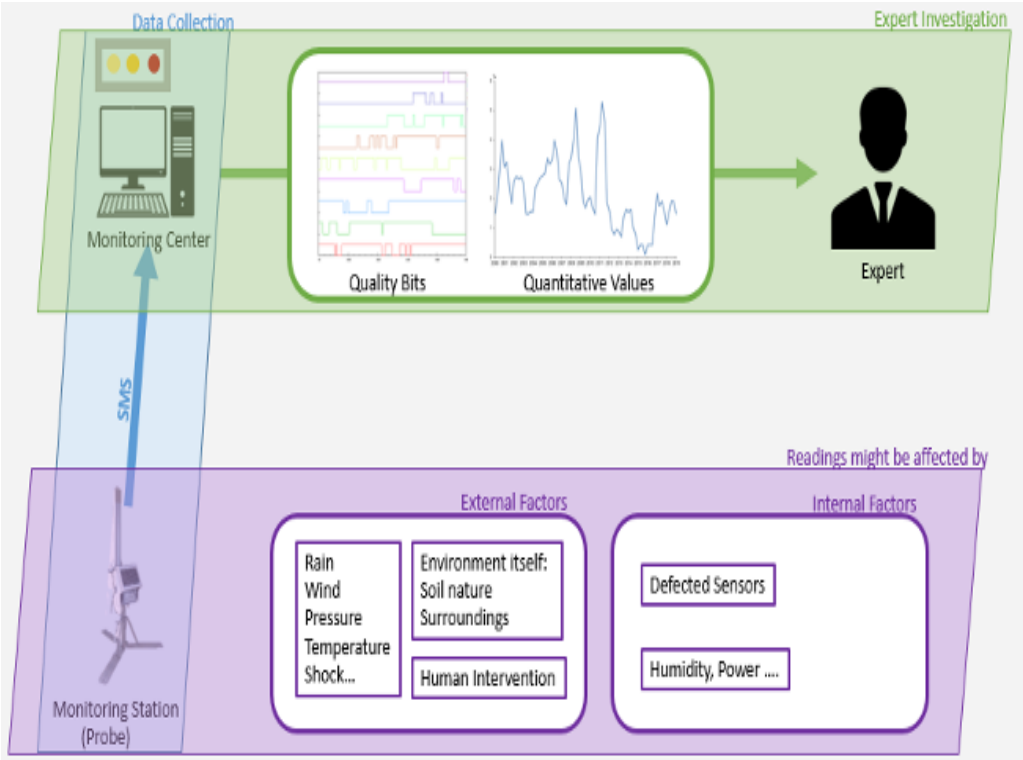

Figure 2.4: Interaction Diagram

As for the Network Architecture, a probe is disposed in each region of interest and they send data to the data center. Moreover, we should mention here that there is no communication between the probes.

2.2 . Role of Experts

After going through the ecosystem of the REWS, we will now focus on the role of the experts and the intelligence performed by them in order to point out the incident identification and the decision taking processes. This will help in understanding clearly the experts' way of investigation in order to remove them later from the loop and introduce the intelligent REWS.

In case of an emergency, the server starts the alarms and sends messages to the experts. There are several steps in order to verify the accuracy of the data. A high persistent gamma dose rate that last more than 3 to 4 hours might require expert to go to the region of interest and identify the cause of the GDR with special equipment. Note that after 6 hours of the first alarm, in case of a real persistent threat, the necessary procedure will be taken.

The purpose of this section is to highlight the role of the experts in identifying the alarm process. Through this section, we will introduce the internal and the external factors that can affect the gamma dose rate values. Then, we will introduce the incident characteristics where we will explain the incident identification process and the incident's shape and duration roles in distinguishing between the different causes behind a high gamma dose rate incident.

2.2.1 . Internal and External Factors

The experts start the investigation process referring to the internal and external factors that may affect the gamma dose rate readings. They need to check multivariate time series data from different sources in order to be able to determine the type of the incident occurring.

- **Internal Factors**

  Internal factors usually depend on the components of the probe and their proper functioning. It could be defect sensors or humidity, the temperature of the probe, the power, or any other component of the probe. Most of the time, the internal factors can be detected through the quality bits readings sent by the probe to the data center.

  Several internal factors, like the GM tubes (as shown in Figure 2.5), the power, the humidity, the balance and so on can affect the probes' readings which might lead to false high gamma dose rate readings. Some of the quality bits can notify the experts about possible damage in the internal component of the probe.

  The scenario described in the Figure 2.5 illustrates how an internal factor can affect the gamma dose rate. For instance, when the low dose tube of the probe is broken, thus it affects the quality of the gamma dose rate value. Its interpretation is no more reliable. This type of scenario will produce a false alarm.

  Moreover, sometimes the quality bits can also identify some external factors. Some readings like rain or temperature can be compared with the region values in order to detect possible human intervention. For example, a probe placed in Beirut in summer shows a temperature equal to 14°C and the weather forecast showed a temperature of 30°C, then experts can presume that someone or something is covering the probe and check it out. Another example could be if the probe is showing rain but in that region there is no rain at that time, it could be that someone is throwing water at the probe.

  The environment can sometimes cause erroneous data. To be more precise, when it snows, the probe and soil may be covered in snow. If in that region the background is about 30%, then when it snows the background's value will decrease. Let's say it became around 20%. In this case scenario, if the GDR increase and became in a few hours 30%, knowing that the snow is

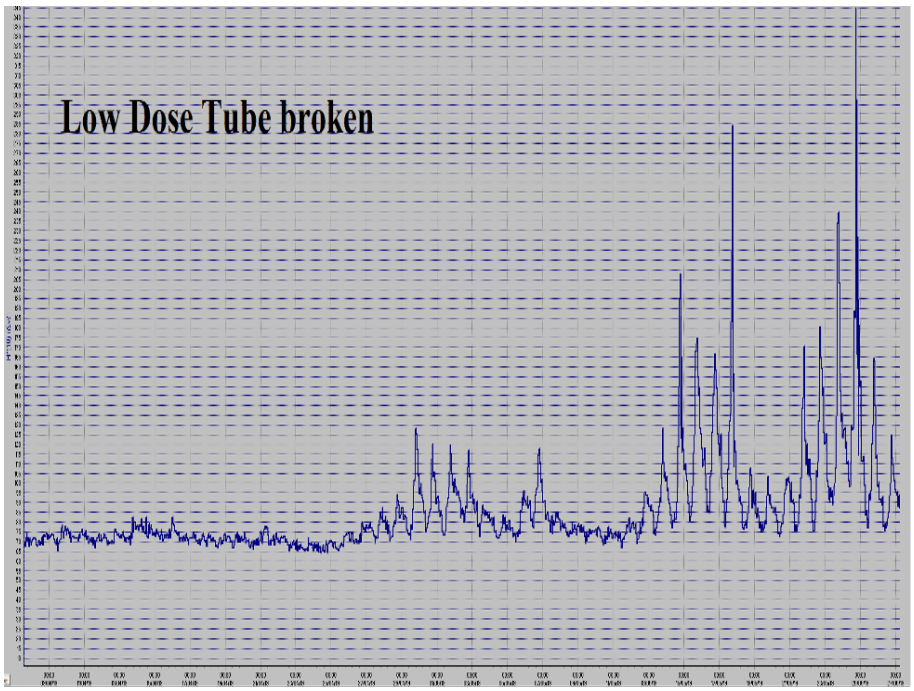

Figure 2.5: Low Dose Tube Broken Effect on Gamma Dose Rate

persistent, this 30% is not equivalent to the initial 30%. It is actually the 1.5 time higher than what it is supposed to be so there might be an undetected problem.

Note that everything that is not a component of the probe and may affect the GDR is considered an external factor.

- **External Factors**

  The weather can play an important role in producing high gamma dose rate readings. In the scenarios respectively described in Figure 2.6 and Figure 2.7, we see how the wind and lightnings directly and immediately impact the gamma dose rate. In these scenarios, we observe many peaks that do not last very long. The wind (as shown in Figure 2.6) can cause the probe to be shacked many times during a specific period of time. This can lead the probe to send high gamma dose rate readings to the data center.

  On the opposite, the rain impacts the gamma dose rate in a completely different manner. The rain (as shown in Figure 2.8) is one of the main factors that cause the GDR to augment. It may or may not provoke radon gas in the air which will cause the GDR to increase importantly. But the persistence will last from two to three hours than it will start decreasing. This is how the experts can validate whether the incident occurring is because of the rain or not. Sometimes, even if it continues to rain, the gamma dose

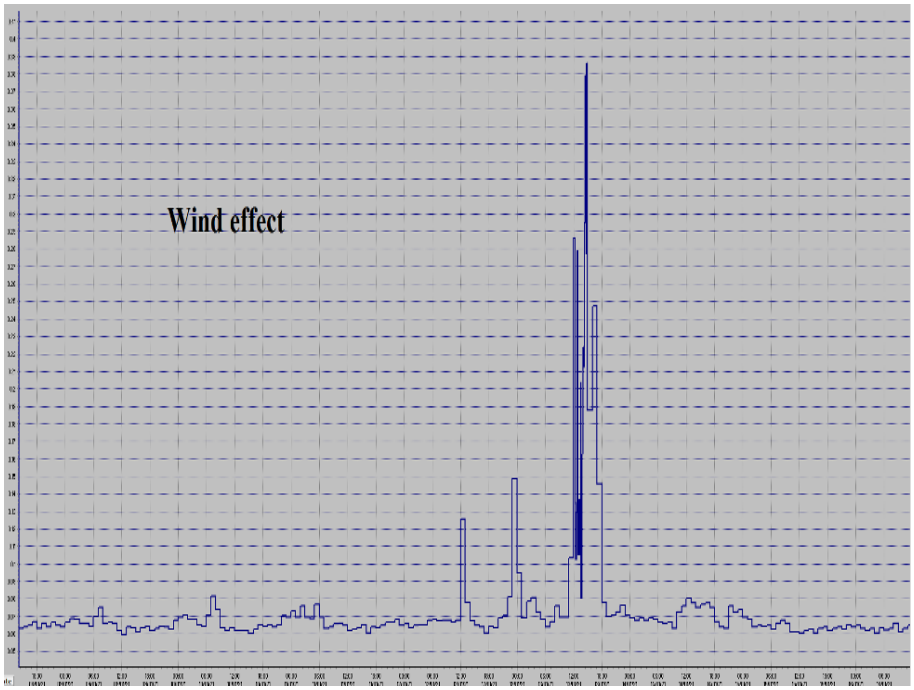

Figure 2.6: Wind Effect on Gamma Dose Rate

rate readings will return to normal values and will not be affected anymore. The effect is seen when the soil is dry, not when the soil is already humid.

Within certain conditions, the wind and the rain effects are considered as innocent. Rain requires the gamma dose rate to start reducing after two or three hours while wind shows short peak values but when it stops the peak values change immediately to become normal values again.

In addition, the pressure is an important factor, since it can affect the concentration of the radioactive elements. Since the pressure differs from a region to another due to the latitude, the pressure also plays a role in the different background between the regions. Note that a change in the pressure can cause the gamma dose rate to increase, but at some extant if it's a normal grow than it is considered innocent.

Last but not least, imported materials may cause a high gamma dose rate that may persist more than two to three hours. In some countries, when imported items or transported radioactive materials passes near a probe, the gamma dose rate value will exceed the peak value. Note that these products while being transported all over the country, so they will affect any region on their path. But since the product are moving their effect in the region of concern will not persist. More importantly, it is critical to mention that in the case of transported product, necessary procedures must be taken to know what are those transported materials and why are they being transported

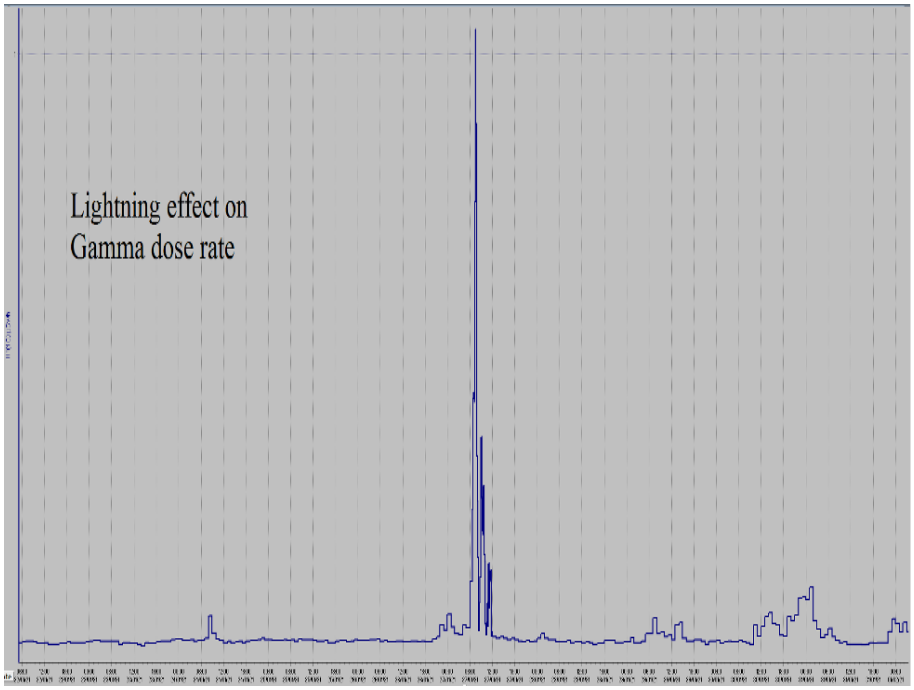

Figure 2.7: Lightning Effect on Gamma Dose Rate

and what is their usage.

The case scenario of an individual going through radiotherapy and passing next to the probe may also cause a high gamma dose rate that is also considered innocent.

2.2.2 . Characterization of the Incidents

1. **Incident Identification**

   Understanding the alarm process caused by an incident was an important task for us. We relied on the parameters used by the experts in order to understand the context of incident identification.

   Experts usually starts analyzing the gamma dose rate time series data (shown in Figure 2.2) once the alarm is triggered in order to identify the current occurring incident. Two important parameters control the alarm process which are the peak and the background values as shown in Figure 2.9. At the beginning of the system, a range of values is fixed by the experts representing the default gamma dose rate values which are the safe values of the GDR in the environment. This range is called the background, that can be different from one location to another. The soil and the surrounding cause by their nature a certain radioactivity. There is a lot of natural element in the air that can cause the gamma radiation. For example, fruits or the buildings play a role in the radioactivity of the area. This radioactivity is normal and

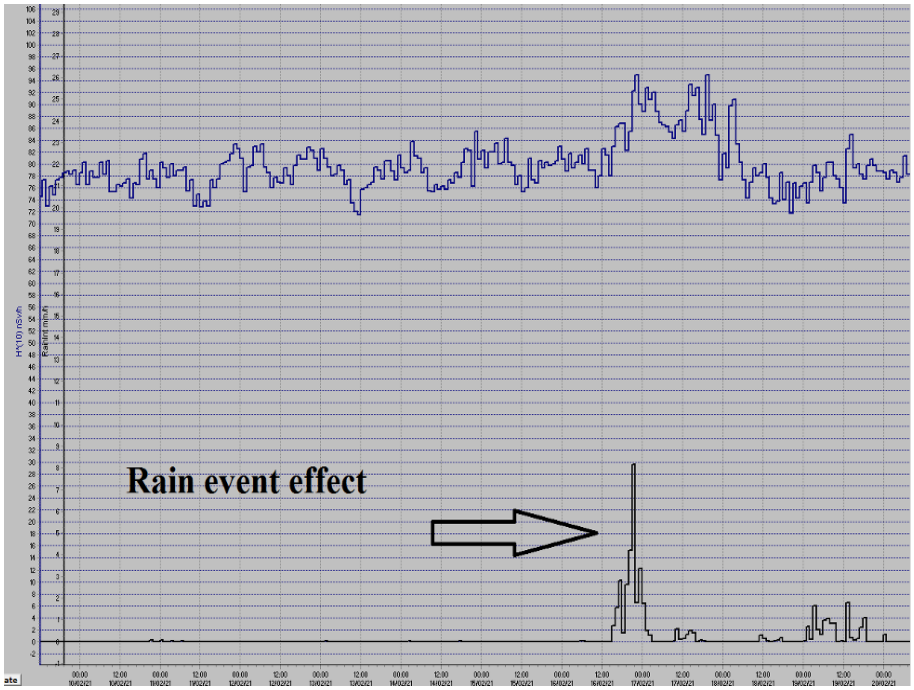

Figure 2.8: Rain Effect on Gamma Dose Rate

is always present. This creates a background that differs from a region to another due to the nature of their soil possibly how near it is to a nuclear accident like Hiroshima [10]. Or they are near a nuclear power plant. This leads to different background level for each region. Moreover, this range is not fixed and can be evolved over time due to the aging of the probe and other environmental factors such as rain, wind, pressure, temperature, etc...

The peak value is the value when the GDR values move from the safe value to an unsafe value fixed earlier by the REWS experts. It is important to mention that setting the peak value depends on the background value and on the experts. In some countries, experts consider that the peak value is equal and greater than 1.5 times the background value. While experts in other countries refer to the values that are equal and greater than 2 or 3 times the background value as peaks [11].

Since the peak values of each region depend on the background value of the region, the experts usually check the background level of a specific region and evaluate the peak values to know if there is a threat or no. The background may vary depending on the season, the temperature, along with many other factors.

In the Figure 2.9, we can notice how the gamma dose rate time series data readings trigger the alarm indicating that an incident is occurring. Once the GDR readings move from the background range (represented by B1

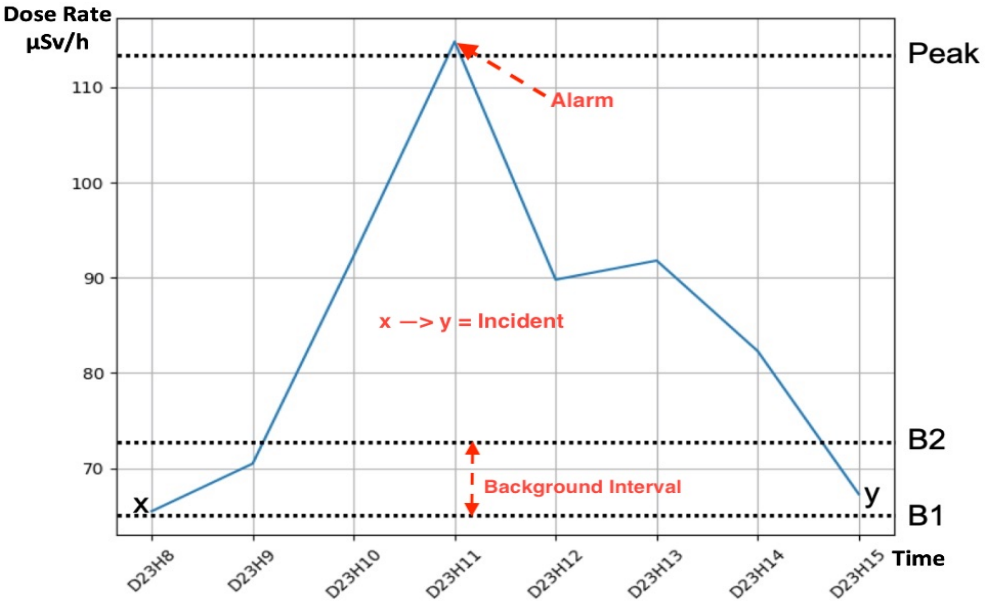

Figure 2.9: Incident Identification Process

and B2 in Figure 2.9) reaching the peak value, an alarm is triggered to notify the experts that an incident is occurring. In Figure 2.9, an incident is characterized by a shapelet extracted points (x and y) in the time series. Thus, experts start their investigations to explore the situation as soon as possible.

2. **Incident Shape and Duration**

   Analyzing the graphical shapes represented by the incidents in the historical databases, we noticed that they are of different shapes and of different durations. Internal and external factors do not affect the gamma dose rate readings in the same way. The shakes caused by the wind, for example, result in a higher gamma dose rate readings than the rain. However, the duration of the incident caused by the wind shakes is too short compared to the one caused by rain. Thus, we classified the incidents as *Hard Parabola* and *Soft Parabola* incidents' graphical shapes as represented in Figures 2.10 and 2.11 respectively. The shape and the duration of the incident play a vital role in characterizing the incident alarm as real, innocent, or false later.

   - **Hard Parabola:** It is one type of the shapes that could be formed by a specific incident. This type of shapes can automatically be referred to quality bits errors or some environmental factors such as the wind, that cause the probe to be shaken for a short period of time, which turn the investigation process to non-risky situation. In a *hard parabola* incident, the gamma dose rate goes up and back down in a relatively

short time. Figure 2.10 shows an extracted *hard parabola* incident where the gamma dose rate took $\leq$ 30 minutes to go down.

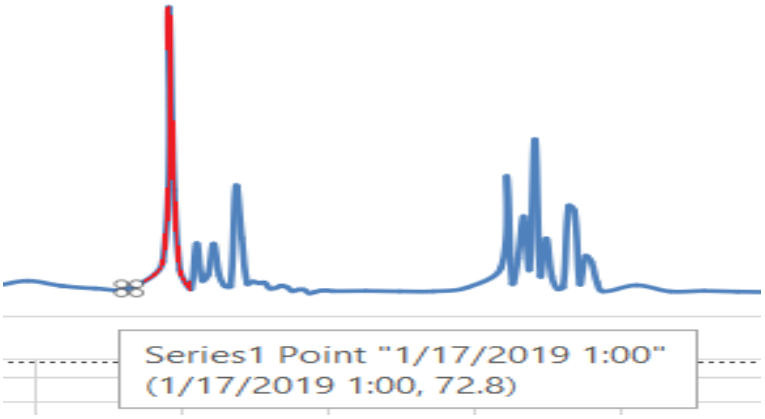

Figure 2.10: Hard Parabola

- **Soft Parabola:** It is another type of the shapes that could be formed by a specific incident where the gamma dose rate goes up and takes a relatively long time to go down. This type of incidents' shapes is considered as very important in our research as it needs closer and deeper investigation process by the experts in order to explain the running situation as soon as possible. It could be either innocent or risky situation that lasts for more than 30 minutes. Figure 2.11, for example, shows a high gamma dose rate incident that took $\approx$ 3 *hours* to go down.

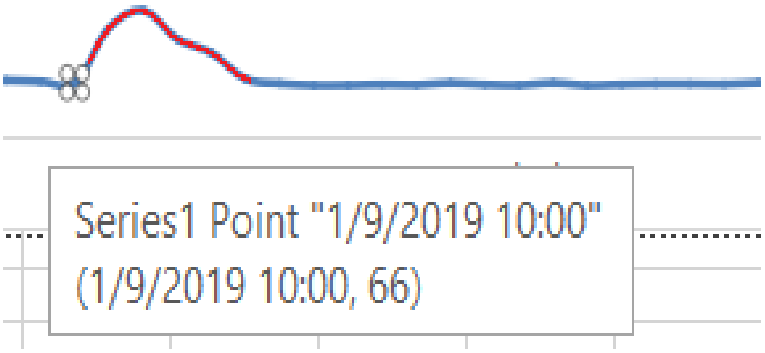

Figure 2.11: Soft Parabola

2.2.3 . Experts' Investigation

During the whole process of this thesis, there was a lot of communication with the Lebanese and the Germany experts in order to capture their knowledge and their ways of investigation upon receiving high gamma dose rate readings.

When an alarm is raised a considerable amount of time and efforts are consumed by the expert to analyze the parameters that are stemming from external or internal data sets (as shown in Figure 2.12) such as weather data sets in order to explore the situation as soon as possible. As there is no automated data collector, the experts must carry out data searching and data fetching operations manually.

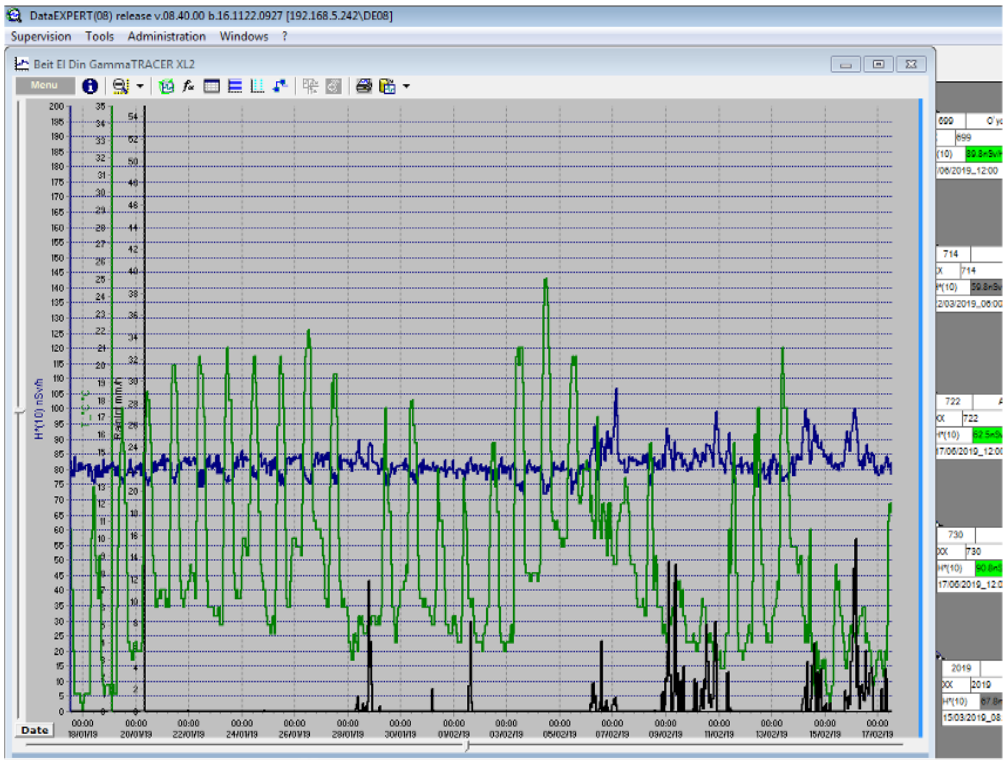

Figure 2.12: Analyzing Different Time Series Data During an Incident

For example, the rain and the wind in a normal case scenario may cause a high gamma dose rate. But during a real incident occurring at a specific region, wind may transport the radioactive materials to other regions where they will persist in the area. And as for rain, a radioactive cloud (that is also transporting radioactive materials) may rain above a certain region causing all the transported radioactive materials to be disposed in the soil of that region. The transported radioactive materials may persist on the soil which will permanently result in high gamma dose rate readings. This may lead to an evacuation of the region as soon as possible. In case of a nuclear accident, some regions were banned.

Note that in a normal case scenario, the GDR goes up, persist for two or three

hours and then start falling to go back to its normal value. Facing a real problem, the GDR will grow and persist more than six hours. Thus, the experts rely on different factors while analyzing the cause behind the incidents.

First they investigate the internal factors to check if the readings are because of any quality bits errors. Next, they introduce data from different data sources (such as weather) to check if any external factor is affecting the current high gamma dose rate readings. Finally, they evaluate the situation in terms of the shape and the duration of the incident where hard parabola shape incidents are eliminated and soft parabola shape incidents are analyzed with further investigation.

Knowing that the internal factors errors result in false alarms, some external factors result in innocent alarms. The innocent alarms cause the gamma dose rate readings to increase but they are still considered safe since the readings will return normal after a specific period of time. Note that in the current system more than 90% of the alarms are either false or innocent alarms since the system is unable to distinguish between the type of the incidents occurring. In other words, every time it rains, the GDR is affected and the whole alarm system goes on for two to three hours.

At the end of the investigation process, the experts will be able to categorize the incoming incident as a false incident (because of internal factors), innocent incident (because of external factors), or real incident.

2.3 . Traditional REWS Shortcomings

Existing REWS solutions have various shortcomings. Our first look at that problem was to analyze the gamma dose rate time series data in order to find possible common behaviors based on the combination of these causes. Being able to identify the behavior of a certain cause may help us analyze rather or not it is the only reason of the high GDR. We want to analyze the gamma dose rate time series data in order to detect the cause as soon as possible in order to take the necessary procedures.

Experts may encounter many scenarios during their work. These scenarios reflect the variety of the situations that occur most of the time. However, multiple factors can be combined together such as rain and wind making the recognition of the cause more complicated.

Moving on to the most critical part: the real case scenarios. When we talk about real case scenario we mean the scenarios that concern a real threat to the population living in the area. Fortunately, real alarms are very rare, but as you can imagine whenever the gamma dose rate readings start to increase and reach the peak value, we can't tell if the readings will continue increasing or decreasing and when they will be back to normal values.

Thus, the most critical shortcoming is the manual intervention of the expert that is heavily time-consuming, labor-intensive, and risk-prone. Most of the time,

the expert cannot classify the alert immediately as he/she needs to wait for further readings of the gamma dose rate to see if it will return to normal. This can take hours due to some parameters such as rain. In addition to that, many data sources may not be available for inspection at just any time which just elongates the useless process of marking the event as not alarming. Therefore, it is not possible to make a faster or real-time inference using the current approach.

Knowing that each country depends on specific parameters when analyzing the data behind the high gamma dose radiation, many data sources should be combined together. Some are collected in a continuous manner by the REWS and stored in an historical database. But many others data sources must be queried on demand when an investigation is launched by an expert. Combining all these heterogeneous data sources on the fly is also a difficult problem in itself. Thus, many information used by the experts during the investigation process, such as the background, incident cause label, and weather data, are not stored in the historical databases.

Moreover, the IAEA (International Atomic Energy Agency) [12] is working on an official template called IRIX (International Radiological Information Exchange). As an end-purpose, all collaborating countries with IAEA must share their data through this template. The only time series data required is the GDR along with the location of the probe. Thus, the GDR time series data are the common type of data monitored and stored by all the countries using the REWS around the world. This highlighted the problem of how to get the same intelligence of the experts with less knowledge. This should be solved and addressed from a computer science point of view in order to introduce a global intelligent tool.

Another dimension of the problem concerns the variability of the peak values that evolve over time and that is also dependent on the location of the probe itself [13] . As said earlier, these predefined peak values are essentially chosen based on observations or experience at the beginning, but they evolve slightly over time on a monthly basis, making the comparison of the time series over multiple months not an easy task. This highlights the difficulty of dealing with the system in terms of preprocessing in order to address the different background issue generated from different probes.

Moreover, the traditional system is a tool that rely on external confidence. The experts are always needed to perform the investigation process and to validate the cause behind the high gamma dose rate readings.

All these examples illustrate the difficulty and the heterogeneity of analyzing the gamma dose rate shape and understanding its causes in order to classify properly the incident in an automatic way. For all these reasons we believe that the research problem is interesting to be tackled as it will require many different techniques or approaches to be used. This is the reason why we define a new framework to offer an end-to-end solution towards an autonomous REWS.

The data used in this work was acquired from the Germany's Radiation Early

Warning System. It is composed of minute-by-minute gamma dose rate data of the past 10 years for over a thousand probes. The data is divided for each probe individually and within each probe's data, each year's data is saved on its own. The data can be visualized as time series as shown previously in the Figure 2.2. The probes in the data are distributed into 9 different divisions each division covering a certain area in Germany. It is important to note that data of probes placed in the same area act similarly as they are affected by the same conditions. For this reason, in our work, we decided to take 45 random selection of probes from the different divisions and combined their data to use as a single source. This sampling doesn't affect the accuracy of our work as we made sure to have data from the different divisions to represent each equally.

2.4 . Intelligent REWS

After introducing the context of the traditional Radiation Early Warning System and the shortcomings behind it it, we moved forward in addressing these shortcomings in order to turn the traditional REWS into an fully automated one. Knowing that the traditional REWS has no intelligence and the intelligence behind it is due to the experts, we conclude our research in introducing a new framework that can tackle the shortcomings and change the behavior of the REWS to an intelligent system. This intelligent system can distinguish between the type of the incidents easily and can predict the cause behind them as soon as possible.

2.4.1 . Objectives and Challenges

The main objective of this research is to develop an end-to-end solution that will be integrated with running REWS systems without any disruption or without replacing it completely. Indeed, before replacing the expert, the system should prove its accuracy to predict the right answer. Thus, a learning process should take place at the beginning until it reaches its full potential and work on its own. Today, we assist to the explosion of machine learning techniques and complex algorithms in order to help experts or non-experts to learn more about their data. Machine learning techniques might help building predictive models in order to have a real-time proactive system. However, in order to apply these techniques, some preliminaries analysis should be done to better characterize the problem that needs to be solved. The main objective of this research is to analyse REWS and see if the expert can be removed from the picture and replaced by an autonomous REWS. There are many challenges to address before reaching this goal. The work described in this thesis is the first attempt to do so, as to our knowledge it does not exist autonomous REWS in the literature.

The ideal method to change the existing system to a fully automated system is to start investigating the historical data presented in order to learn from it and discover hidden patterns. As the current process relay on the experts' experience

in analyzing the real time measured values, the main challenge could be introduced as changing the experts' knowledge into a computed process that can identify the risky situations as soon as possible.

The current process can be considered a complicated process with high cost before classifying the peak situation as a real, innocent, or false alarm. This requires several investigations regarding weather forecast data, system components defects, or other causes that are leading to specific peak values that are 1.5 times (or more) the background value obtained in normal situations. Although experts trigger several external and internal data sources during their investigation process, however, the data that are triggered from external data sources are not stored in the historical databases. The only values kept in the historical databases are the gamma dose rate readings and the quality bits values. In addition to that, neither the background values nor the peak values are stored in the historical databases knowing that these values can change over months based on the aging of the sensor (probe) and climate change. This will raise the challenge of using a limited source of information to move to a full intelligent system instead of relying on all information triggered by the experts during alarm analysis. Thus, this highlights the challenge of moving from multivariate time series analysis to univariate time series analysis.

Although the objectives of our research is to propose a fully automated framework, we strongly believe that at the initial stage the solution needs an expert opinion to validate the results produced by the system. This validation is important due to the sensitivity of the use cases that will be implemented using this solution. This will help in increasing the accuracy rate of the proposed framework. However, in case of exceptional use cases that were not known, the involvement of the experts would help to enhance the solution by training the framework over the data and make it capable of recognizing incident causes that were unknown before.

Moreover, another aspect of the challenges will be the experts' confidence challenge. Knowing that the radiation early warning system is a system that is used in a critical field, gaining the experts' confidence in a new intelligent system was not easy. Thus, we need to introduce the experts during the development of every stage to make sure that they confirm the results behind each stage and the overall outcome of the proposed intelligent system.

2.4.2 . Refining Incoming Monitored Incidents (RIMI)

Through this section, we present our RIMI framework (Refining Incoming Monitored Incidents) that highlights the different steps that are needed to take place before reaching an autonomous REWS solution. In this framework, we plan to develop a list of components from data acquisition and normalization, to building a predictive model on a real data set produced by a running REWS, then by using it to predict the right classification of the alarm on real-time data.

Developing this framework, it was clear for us that we need an offline model

and an online model. The offline model is used to learn from the data to find possible patterns. Therefore, we need to do a context evaluation where machine learning algorithms and techniques will be applied so that the provided patterns will be used by the online model.

On the other hand, the online model is the one that will be integrated into the system and will be processing the current data stream. It will have also a context evaluation of the current context. The machine learning model will be applied in order to categorize the incoming incidents based on the results obtained in the offline model.

Hence, the proposed framework was the tool of defining the two problem statements behind this thesis.

2.4.3 . RIMI Framework Architecture

The framework consists of two main phases: (1) the building of the predictive model and (2) the online detection and prediction. Figure 2.13 illustrates more in detail each of the main components. As the incident is caused by a high gamma dose rate level which can be harmful for humans and environment, this framework aims to replace a human-driven verification system that refines the incoming incidents and alerts and detect its cause by doing it automatically with a high level of accuracy.

1. **Building the Predictive Model**
   To build the predictive model, it was essential for us that we need to explore the existing incidents in the historical databases. These incidents should be extracted and distributed into groups based on their common behavior. Thus, these groups should represent the causes that are leading to the high gamma dose rate incidents. To obtain these groups of incidents, we investigated the time series clustering models and applied them on our data.

   We started the first phase by extracting the incidents from the historical databases. After analyzing the extracted incidents, we noticed that the background level is not the same at all probes because of the location and environmental factors which means a peak value at one probe may be a normal value at another one. So, this results in incidents with different levels and different scales because of different factors affecting the high gamma radiation. Also, these factors can result in short-time or long-time incidents which cause a big variation in the length which can reach 100 or 200 points. So, in order to help experts to distinguish these incidents, we need to group them into similar groups and learn from them. But we found ourselves facing unlabeled data, so we raised the idea of using time series clustering that can deal with unsupervised learning.

   From here, a research question was actually raised "What is the best-fit machine learning model that should be used for clustering unlabeled time series data of varying lengths and different scales?"

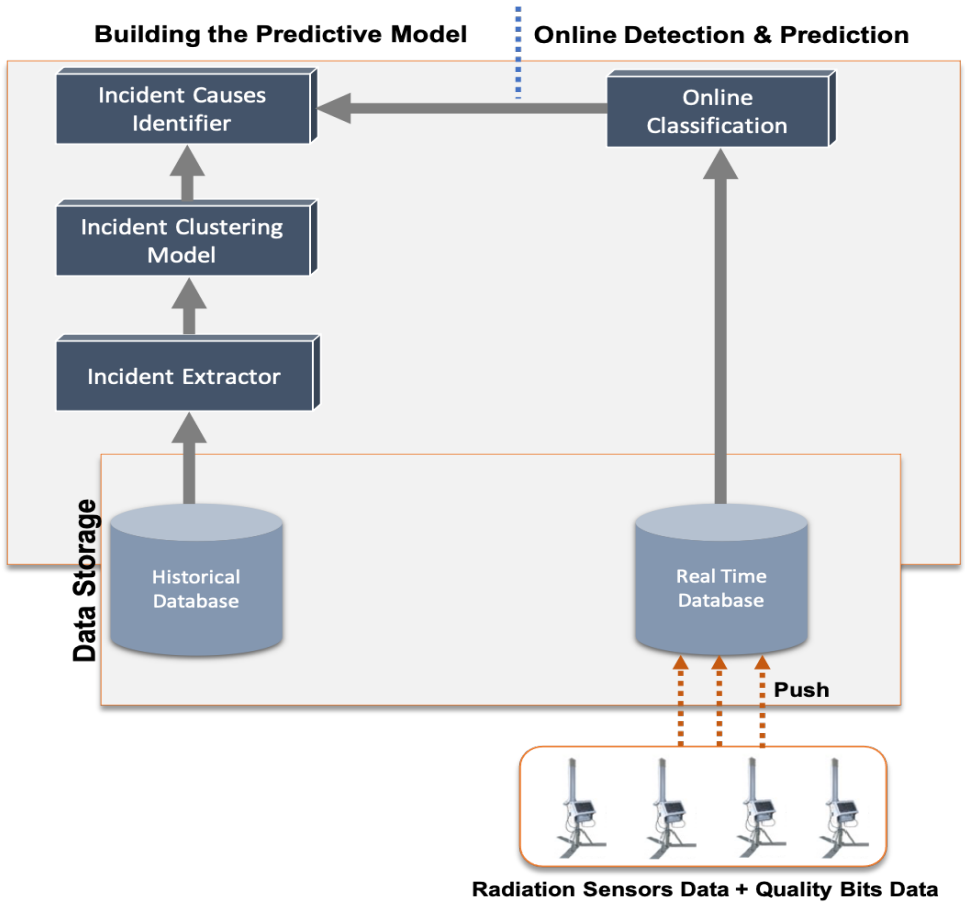

Figure 2.13: A Framework for Real-time Radiation Detection (RIMI Framework)

2. **Online Detection and Prediction**
   After developing the "Building the Predictive Model" phase, we moved to the second phase of the RIMI framework which is the "Online Detection and Prediction" phase. The main objective of this phase was to capture the incoming incident at real time and to try to match it to one of the clusters that were the result of the first phase. We noticed through this phase that we are dealing with a time series classification problem.

   The online phase will be combining different challenges at the same time. The first challenge will be analyzing the incoming radiation data and searching for peaks. However, the second challenge will be categorizing the real time incidents as soon as possible. This problem needs to be taken into consideration along with another challenge which is the variation of the incoming incidents with respect to the scale and length. Upon peak detection, the respective incident should be compared and matched to other incidents

in the predefined clusters. Thus, we raised the idea of using time series classification.

From here, a research question was actually raised "What is the best-fit machine learning model that should be used for classifying online time series data of varying lengths and different scales?"

Moreover, we dedicated two chapters in this thesis each one of them is tackling each phase of the proposed framework.

3 - Building the Predictive Model for RIMI Framework

Contents

For a long time now, time series data analysis has been a center of attention in research as it is used in the different applications such as weather prediction [14], motion capture processing [15], analyzing insect behavior [16], pattern discovery on health-care data [17], and so on. Similarly, in the radiation monitoring domain, intelligence can be extracted from the gamma dose rate time series data.

In chapter 2, we introduced an end-to-end framework towards an intelligent Radiation Early Warning System. By analyzing the incoming real-time gamma dose rate time series data, a supervised learning model shall be able to recognize the reasons behind high gamma dose rate incidents using historical data with as little human intervention as possible.

In order to properly identify the reasons behind the incidents the model has to have as input a data set of historical incidents labeled by the reasons behind them. However, the experts' evaluation of the incidents are not maintained in the historical data; the only data maintained in the databases is the gamma dose rate time series data of the different probes. Thus, we are dealing with a univariate time series data. Now, with the proliferation of the use of Artificial Intelligence and Machine Learning specifically to automate processes in the different domains and sectors, instead of manually extracting and labeling the historical incidents, an interesting solution would be to introduce an intelligent model that extracts knowledge (behavior patterns) from the historical gamma dose rate data present in the databases to identify incidents and their underlying events and label them with as little human intervention as possible.

A trivial approach for identifying the cause behind a certain turbulence in a time series data is to assume that a certain familiar event results in a recognizable temporal trace in the data, and then scan the time series data for such traces. For example, in the case that events can be recognized by a certain pre-known geometrical shape or by certain properties such as the amplitude or the duration.

However, firstly, the data that we are dealing with are not labeled incidents to be able to inspect them and make such claims. In fact, what we have here is not data sets that have been acquired by experimentation and properly labeled by experts; the data considered in our research are raw time series data coming from the historical databases of any gamma dose rate probe from around the world that could be affected by anything. The data' characteristics have to by analyzed so that the incidents can be extracted, processed, and analyzed by the proposed framework as autonomously and with as little prior information as possible.

Secondly, incidents caused by the same event may not have a recognizable temporal trace or characteristics but more of a common behaviour. For example, a certain event may cause peaks of increasing amplitudes that decrease over a longer period of time, another may cause an abrupt increase and maintain its amplitude for a period of time and so on.. Note that incidents caused by the same event can

last for a varying length of time and reach different amplitudes.

Most of the work in the literature in the different research domains of health, electricity, environment.. are done on specific data sets that have certain characteristics because of being done in an experimental environment for research purposes. However, what we want to achieve in this chapter is to extract intelligence from the raw historical data.

Throughout our work we dive into the field of unsupervised time series clustering as it requires no labeled data. Specifically, we study the clustering of time series of varying length as we found it was a characteristic not properly addressed in the literature. Moreover, we researched the different state of art approaches and evaluated their compatibility with our data set. Knowing that we are not contributing to the clustering algorithms, however we are proposing a kind of methodology where we are fine tuning the process of seeking for the best way to do clustering. This led us to the hardest part of our research where we tested all types of combinations between similarity measures and clustering algorithms. In the end, we present our contribution which is the machine learning model we introduced that achieved the best results through testing.

3.1 . Unsupervised Machine Learning Model for Time Series Clustering

In this chapter we propose an unsupervised machine-learning based framework to automate the process of extracting high gamma dose rate incidents from the raw historical data present in the REWS databases and identifying the underlying event behind each. Our approach is to group similar-behaving incidents as caused by similar events to label them with the help of the experts so that the resulting groups can be used later to autonomously identify the reasons behind incoming incidents in real time so as to decrease the time and efforts spent as well as to increase the efficiency and accuracy of the process.

In our work, we are dealing with the raw historical time series data, so we first present the incident extraction process that was devised with the help of experts to suit their investigation criteria. After studying the extracted incidents and their characteristics, we explore the different preprocessing algorithms that fit our purpose and apply the proper ones before proceeding. We, lastly, introduce our machine learning model for automating the grouping of the incidents by the different underlying events and evaluate the results with the help of the experts.

As we aim to group unlabeled incidents with as little human intervention as possible, our research is in the field of the unsupervised machine learning time series clustering.

Our proposed framework is divided into three phases which can be seen in the Figure 3.1 and briefly explained as follows:

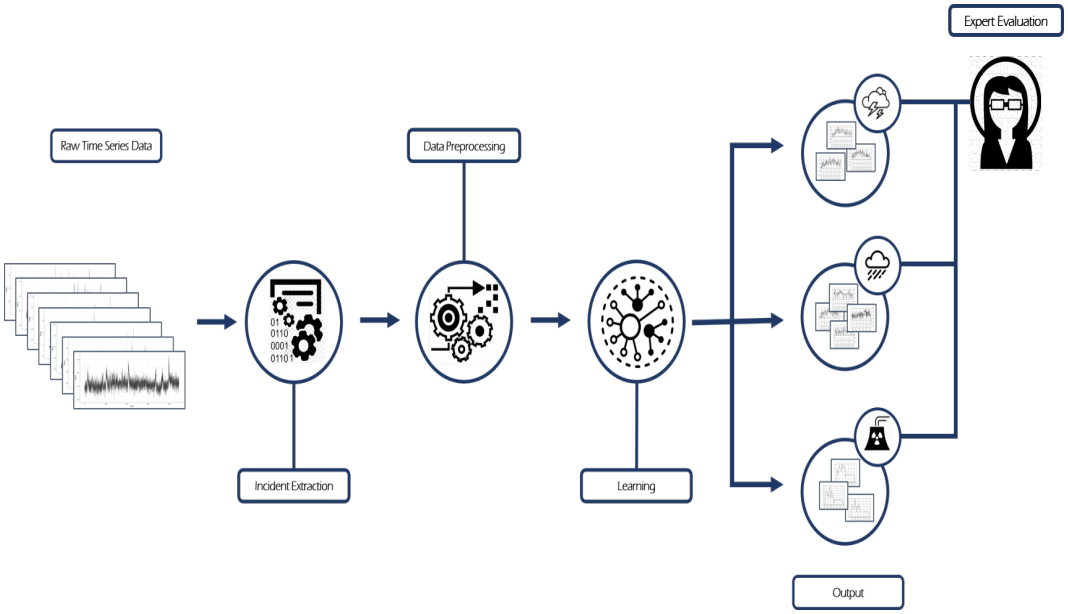

Figure 3.1: Proposed Framework Architecture

- **Incident Extraction and Preprocessing:** Alarming incidents are extracted from the raw historical time series data by autonomously identifying the proper parameters. The raw extracted incidents are studied to perform the proper preprocessing so as to have a data set ready to be analyzed by the multiple algorithms considered in the next phase
- **Time Series Clustering:** The machine learning model is designed to separate the incidents into the different groups (clusters) signifying the different reasons behind the incidents.
- **Cluster Evaluation:** The resulting clusters are evaluated with the help of experts to determine their value and identify the events behind each cluster of incidents.

A main contribution in our work is researching the techniques used specifically when dealing with a data set of varying length samples. The length of the extracted incidents, depending on the event behind it, varied a lot, which had us looking into time series clustering of data samples of varying length were we found that it was specifically properly addressed no where in the literature.

The main research question of this chapter hence could be formulated as follows:

"What is the best-fit unsupervised machine learning model that should be used for clustering time series data of varying lengths and different scales?"

In that direction, we present a comparative study between the different (time series) preprocessing and clustering techniques that are available. We compare

these techniques based on applicability and effectiveness. Besides conducting a literature study, we also apply the different techniques on our data set to test their performance, and when none of the approaches give satisfactory results, we propose a new improved model which gave the best results through our testing.

3.1.1 . Generic Approach

In this section, we briefly recall the three main phases for defining a time-series clustering approach. First, there is the time-series data preprocessing. Second, the similarity measure should be chosen. Finally, the clustering algorithm must be selected. Based on these, we enumerate the approaches proposed in the state of the art for univariate time-series clustering, especially those for the ones of varying length.

Time Series Data Preprocessing

Data normalization and missing data imputation are the basic data preprocessing done on time series data. Both kinds of techniques have a significant impact on the performance of a model, and they should be chosen based on the problem and model at hand.

1. **Missing data imputation:** Missing values cause problems for machine learning algorithms as they will perform better with complete well-formed data. Some of the most popular approaches to deal with this problem are, dropping rows with missing values, statistical imputation, and model imputation.

    - **Dropping rows with missing values** A quick and easy way to imputing is to drop the "offending" entries. This approach is usually a go-to when the missing data is so at random, and the dataset is relatively big so dropping some data points will not affect the accuracy and generalize-ability of the model built. On the other hand, depending on the reason behind the missing data and the size of missing data relative to the data set, dropping data missing rows might result in a significant bias in the model.
    - **Statistical imputation** Another popular approach to dealing with scattered missing data is to impute by replacing the missing value with a statistical parameter such as the mean, mode, or median. Such an imputation algorithm is useful when we cannot afford dropping rows with missing values and the data is normally distributed. In such cases imputing with statistical parameters will not affect the model statistically. However, if the number of missing data is relatively big, then such an approach will affect the accuracy of the model. Also, this

approach adds no novel information and just increases the size of the data set.

- **Model imputation** If the former two approaches cannot be applied to the data set. An interesting approach is to go all out for this phase and apply a machine learning model to the data set. Using the complete records, the machine learning model will be able to impute the missing values. As an advantage, the imputation approach is able to impute the missing data while significantly avoiding any alteration to the standard deviation or the distribution. However, like in the statistical imputation, the imputed data points introduce no new value but only increases the sample size.

2. **Normalization:** The most common normalization methods used during data transformation include min-max, decimal scaling, and z- normalization:

   - **Min-max** normalizes the values of an attribute according to its minimum and maximum values. The main problem of using the min-max normalization method in time series forecast is that the minimum and maximum values of some data set cannot be known.
   - **Decimal Scaling** moves the decimal point of the values of an attribute according to its maximum absolute value. This method also depends on knowing the maximum values of a time series.
   - **Standardization (z-normalization)** in contrast, makes no use of the minimum and maximum values. The data values are normalized by calculating the z-score using the mean and standard deviation of the original data values. This method is also called Zero-Mean normalization because after this, the series is centered around zero with a standard deviation between -1 and 1 as can be seen in the Figures 3.2 and 3.3.

Similarity Measure

Similarity measures are algorithms used to determine the resemblance between different samples. In time series clustering it is the determining factor used by the clustering algorithm to decide which cluster each sample belongs to. Shape-based distances evaluate the similarity of samples based on the actual or the normalized values while feature-based distances evaluate similarity based on extracted features. In our work, we are only interested in shape-based measures. They fall into one of two categories: Lock-step measures, or elastic measures.

1. **Lock-step measures**

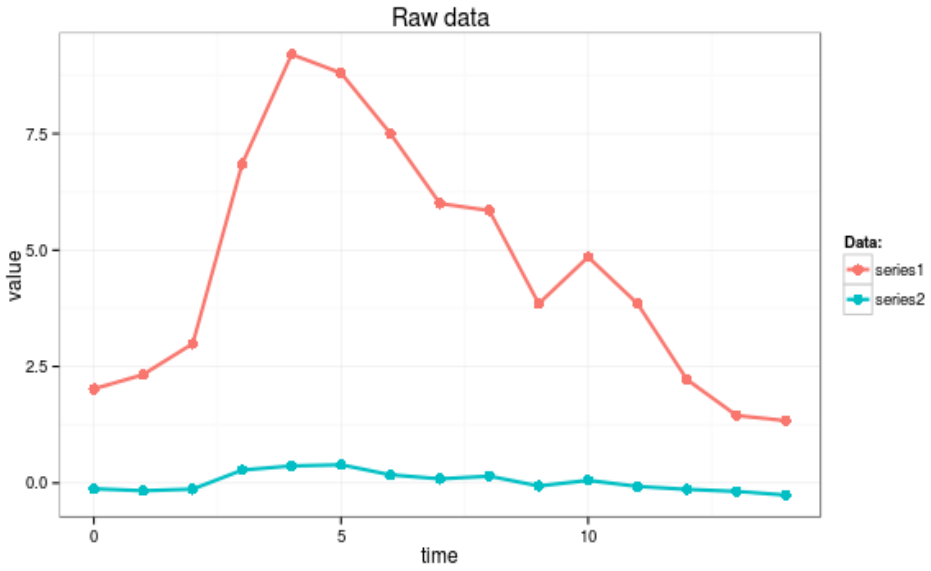

Figure 3.2: Raw data

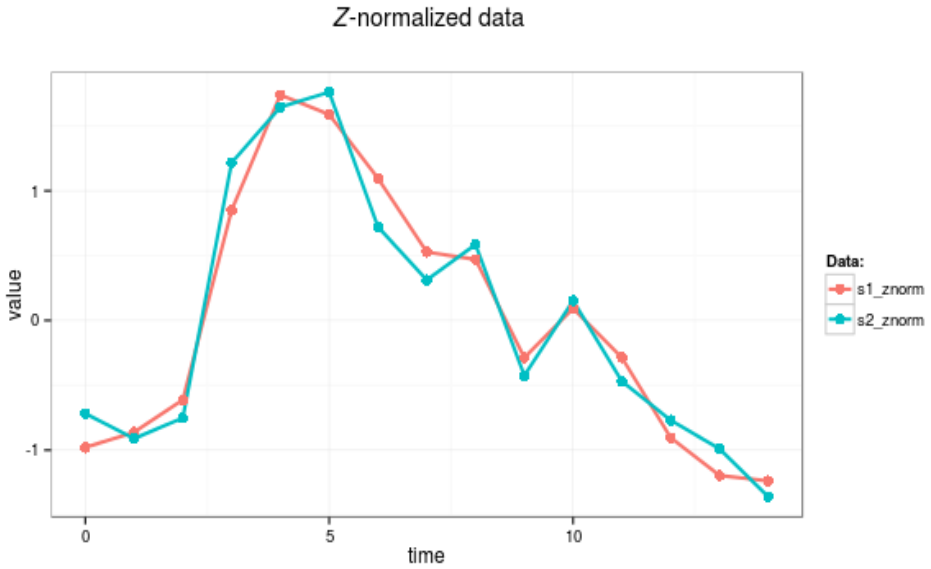

Figure 3.3: Standardized data

Lock-step measures are metrics that evaluate the distance between two time series sequences as the overall difference between each point and its counterpart in the other sequence. These measures require data sequences to be of equal length.

- **Minkowski (L_p norm) distances**, specifically Euclidean [18] , are the most favored lock-step metrics in machine learning. Their popularity is derived from their simplicity and success in machine learning literature as well as their being parameter-free. The Minkowski distance is the L_p-norm of the difference between two vectors of equal length ($n = m$). It is the generalization of the Euclidean ($p = 2$), Manhattan ($p = 1$), and Chebyshev ($p = \infty$) distances.

2. **Elastic measures**

   Elastic measures, on the other hand, provide better flexibility as they permit one-to-many and one-to-none point evaluation. Due to this flexibility, these measures provide better comparison. This flexibility, however, comes at the price of increased time complexity.

   - **Dynamic Time Warping (DTW)**[19], most famous elastic measure in literature, is an elastic measure that was introduced specifically for time series analysis. As the name suggests, it warps the two considered sequences in time in order to deal with time shift and speed variations.

     DTW can be used as a similarity measure between samples of varying length, as it produces a scale-like-effect, stretching and contracting, by accepting many-to-one matching however this also makes it sensitive to outliers. However, it does not satisfy the triangle inequality, even when the local distance measure is a metric [20].
   - **Longest Common Sub-sequence (LCSS)** [21] , is another distance measure that was introduced first for text analysis but has lately been seen in the time series analysis literature. LCSS is calculated by searching for common subsequences in the two sequences which might not occur at the same time but preserving their order.

     LCSS finds the optimal alignment between two series by inserting gaps to find the greatest number of matching pairs as shown in Figures 3.4 and 3.5.

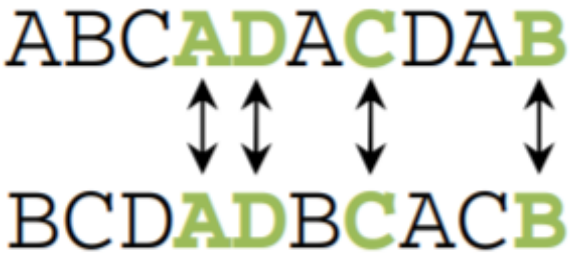

Figure 3.4: LCSS distance with no elasticity

Figure 3.5: LCSS distance with full elasticity

LCSS ensures exact subsequence matches and is less sensitive to outliers as it tolerates one-to-none matching.

Clustering Algorithm

It is concluded by Liao[22] that for time series clustering, many general purpose clustering methods can be applied and the choice of the distance measure is more important than the choice of the clustering method. We therefore only consider hierarchical and partitional clustering methods, as they are the most commonly used clustering methods in literature on time series clustering [22] [23] [24] [25].

1. **Hierarchical clustering**

   Hierarchical clustering [26] takes no parameters other than the linkage criteria, and the most commonly used criteria are the: single, average, and complete linkage [27]. Depending on the linkage criteria a tree-like nested "hierarchy" of clusters is built which can be visualized by a dendrogram. The cluster growing method can be increasing (agglomerative clustering or bottom-up) or decreasing (divisive clustering or top-down) at each step.

   Hierarchical clustering's main advantage is that it doesn't require the number of clusters as input. Once the dendrogram is obtained, the clusters can be

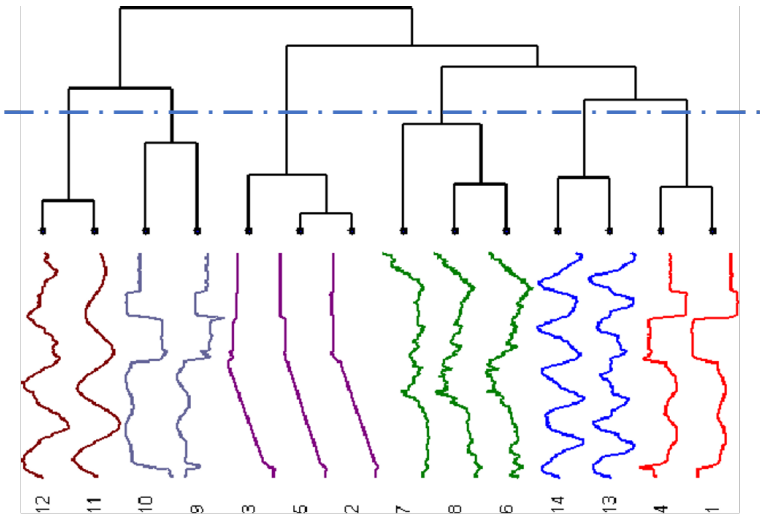

Figure 3.6: Example of a dendrogram for a hierarchical clustering.

decided by making a cut at a certain point. As seen in the example in Figure 3.6 a cut is added for $k = 6$ resulting in clusters: $\{11, 12\}$, $\{9, 10\}$, $\{2, 3, 5\}$, $\{6, 7, 8\}$, $\{13, 14\}$, $\{1, 4\}$.

On the other hand, it requires the distance matrix of all possible pairs of observations. This makes it very computationally expensive and not a favorable option for huge data sets, as the number of elements in this matrix grows with $O(N^2)$.

2. **Partitional clustering**

   Partitional clustering, as the name implies, partitions the data into *k* different clusters where *k* is specified *a priori*. Partitional clustering's aim is to minimize intra-cluster distance and maximize the inter-cluster distance. Partitional methods need the number of clusters *k* a priori.

   K-means [28] and K-medoids [29] heuristics are considered the front-men of the partitional methods. They are both based on the concept of finding the best cluster centers, minimizing the distance between each observation and the center of the cluster it is assigned to.

   - **K-means:** Once the number *k* is decided, the next step is to initialize *k* cluster centers. For an extensive overview and comparison of initialization methods we refer to the work of Celebi et al. [30]. The third step in the k-means algorithm is the while-loop that assigns each observation to the cluster whose center it has the shortest distance to. After going over all observations the new cluster centers are evaluated as the mean of the clusters' assigned observations. The process is

repeated until either convergence or maximum number of iterations is reached.

- **K-medoids:** K-medoid is similar to K-means in the process it goes through to produce the *k* clusters. However, the main difference between the two approaches, is that K-means calculates each artificial centroid (cluster center) using the cluster's assigned observations while in K-medoids the cluster centers are actual observations in clusters (medoids). Another difference within the loop assigning the observations is that the distances needed to calculate the medoid are the ones between the observations, so they can be determined prior to running the algorithm.

  This group of algorithms, partitional, give better results when the data set's time series samples are of equal length because the concept of cluster centers becomes tricky when the same cluster contains time series of unequal length [22]. Choosing the optimal length of the centroid is not by far an easy task. It has to be as concise as possible; short, but also long enough to adequately represent the data it covers [31].

  From all these methods, only the k-means class of algorithms can scale linearly with the size of the datasets.

Determining optimal number of clusters

Clustering methods require the number of clusters k as an input parameter in order to return a clustering. Non-hierarchical methods usually require k to be specified beforehand, whereas for hierarchical methods the value of k can be set afterwards. Two of the main statistical approaches used for evaluation of optimal number of clusters are:

- **Elbow Method:** Is a method that estimates the number of clusters by comparing the within cluster dispersion. In order to determine the optimal number of clusters the sum of squared error (SSE) for each k has to be calculated. Then, a line chart is plotted of the SSE for each value of k. If the line chart looks like an arm, then the "elbow" on the arm is the value of k that is the best. The idea is that we want a small SSE, but that the SSE tends to decrease toward 0 as we increase k.

- **Silhouette Method:** The Silhouette index is proposed by Kaufman et al. [26] and is based on compactness and separation of clusters. It starts by measuring the silhouette score of each observation which is calculated using the average distance from the observation to all fellow observations in the

same cluster and the average outer-cluster distances the observation to the observations of the closest cluster.

A silhouette value close to 1 indicates that the observation is assigned to the appropriate cluster, whereas a silhouette value near -1 indicates that the observation should be assigned to another cluster. The "optimal" value of k, according to the Silhouette index, is the k that maximizes the value of silhouette width.

3.1.2 . State of the Art

A similar approach to what we are proposing was conducted by Kang et al.[32] while studying the weak-wind ABL data. They wanted to discover the underlying physical mechanisms that were causing the variation in its temporal data. The approach proposed was divided into two steps: first to extract the incidents from the time series, and then to try and understand the underlying processes. They took such an approach as it was found that many of the latter processes are unknown.

In their work, they applied a feature-based hierarchical clustering method [33] using a feature-based similarity measure. The similarity measure used was the Euclidean distance calculated between the extracted feature vectors of the incidents instead of the raw data. After clustering, using the dendrogram visualization, they were able to have a good idea about the incidents and their characteristics. In the end, they still had to choose the number of clusters.

Through their work, they dealt with a similar problem to what we are trying to solve. However, first, their extraction method and the fact they went for a feature-based approach make their whole approach domain dependent specifically for weak-wind ABL data. Secondly, they used Euclidean distance which cannot be applied directly on our data set observations because they are of varying length and we are not looking for point by point comparison but more of a behavior comparison. Lastly, using the hierarchical clustering algorithm is extremely computationally expensive and we would have to exhaust all other possible clustering algorithms before choosing to go by it.

In Table 3.1, we summarize the main approaches proposed in the literature to cluster time-series of varying lengths. We found that the most favored similarity measure is DTW, and the most popular clustering algorithm is K-medoids. Combining DTW and K-means does not give valid clusters as stated in [34]. The only approach using DTW and K-Means is proposed by Petitjean et al. [31] who introduced a global averaging method called DTW barycenter averaging (DBA) which is a heuristic strategy. However, combining DTW with k-means seems to have a lot of complications, and even with the DBA averaging method, the verdict is left for the testing to see how the DBA fairs with a big length difference compared with the DTW with the k-medoids model.

Looking into the literature of time series clustering, we found little to no mention of clustering temporal data of varying length. It is true that there is mention

Table 3.1: Combined Techniques in the Literature

Similarity Measure	Clustering Algorithm	Literature
DTW	K-means (DBA)	Zhang et al., 2015[35]
	K-medoids	Liao et al., 2002[36] Liao et al., 2006[37] Hautamaki et al., 2008[38] Gao et al., 2020 [39]
LCSS	K-medoids	Soleimany et al., 2019 [40]

of similarity measures that "work well" with time series of different length however no work has been done to address this specific issue within the clustering models.

Ratanamahatana and Keogh [41] stated that using DTW to compare sequences of different lengths, simple re-interpolation of the data to be of the same length can solve the problem with no significant difference in accuracy. Which is why Thuy et al. [42], in their work, found their requirement that the data set to be of equal length is reasonable as well as they considered it "easy"to perform *homothetic transformation* on the subsequences of different length before comparing them.

While this sounds good for the similarity measure (DTW), it is still not clear if this is still true when the similarity measure is used within a machine learning model. In a recent work, Tan et al.[43] explain that there was a little work done in the literature on the classification of time series of varying lengths compared to the "time-warping" problem. They say the problem is comparatively "understudied and unappreciated". When looking at the UCR archive [44], we see also that there are a lot of datasets that are uniform and not much of varying length only very recently in 2018. That is why we believe that the context of our research will help to have a better understanding of the problem. Unfortunately, due to the nature of the data (radiation level), they cannot be rendered public to the UCR archive.

K-shape was also introduced recently by Paparrizos and Gravano [45] and is considered a novel algorithm in the field of time series clustering. It's scalable and gave competitive results when put against the strongest classic and state of the art algorithms. K-shape is a shape-based clustering algorithm and it has its own distance measure called Shape-based distance (SBD) [45]. The centroid of the clusters in the K-shape algorithm are computed based on the characteristics of its similarity measure.

In their testing, they found that K-shape outperformed all other partitional and hierarchical methods except k-medoids with constrained DTW (cDTW) which gave similar results. However, they discussed how the latter method is not scalable, and how the distance measure requires expert tuning.

However, K-shape requires the samples to be of equal length (or close to it) which is why it cannot be applied on our data (as it is).

To conclude non of the models proposed in the literature addresses the problem of analyzing time series of varying length specifically. Also, no work in the literature has been done on how to cluster time series of varying length. Some algorithms claim that they tolerate samples of different length (DTW, LCSS, k-medoids, etc.), but the question is how well do they work with such data sets within the models, and if none of the models give satisfactory results, how to address that.

3.2 . Proposed Approach for Clustering Gamma Dose Rate Incidents

In this section, we describe the different choices and combinations made in our model to cluster gamma dose rate time-series. Then we will give the methodology of the various experimentation that leads us to our proposal. Our approach is depicted in Figure 3.7.

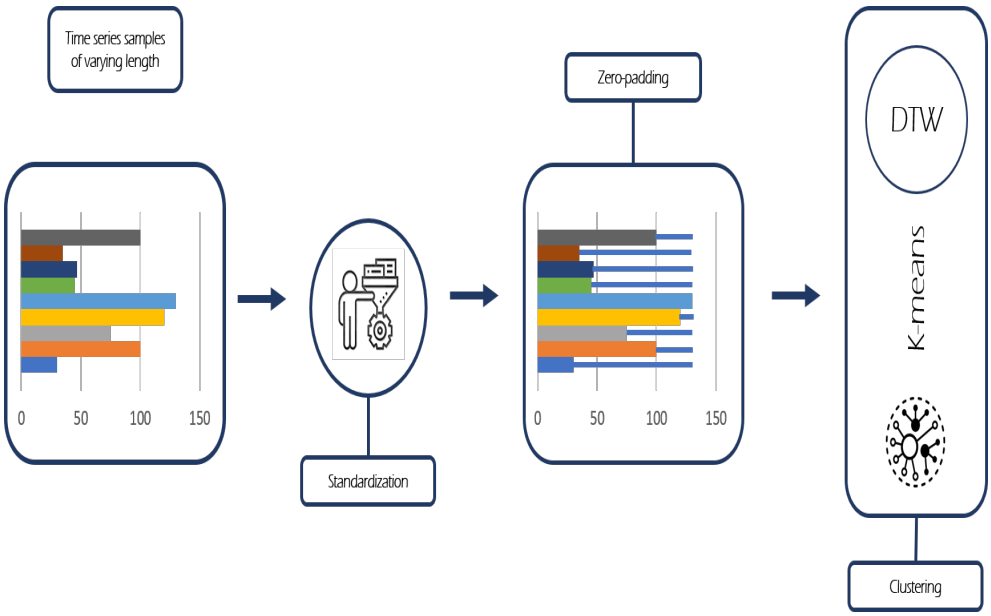

Figure 3.7: Specific Model for Clustering Gamma Dose Rate Incidents

3.2.1 . Incidents Extraction and Preprocessing

As explained before, machine learning algorithms achieve better performance if the time series data has a consistent scale or distribution. Thus, an important attention has been on incident extraction and data preprocessing. Then, we detail the similarity measure and the clustering algorithm we retain. In what follows we present our approach to extract the alarm-triggering-event fragments from historical data automatically with no user intervention.

To identify the *peak* threshold used for incident extraction we applied the concept of sliding window by calculating it each month in order to stay up to

date with the evolving normal range of background gamma dose rate. In order to apply the *sliding window* concept, each yearly time series data was divided into the set of different months subsequences. For each subsequence, the parameters are calculated, the set of values above the *peak* are identified, and corresponding incidents are extracted as shown in Figure 3.8.

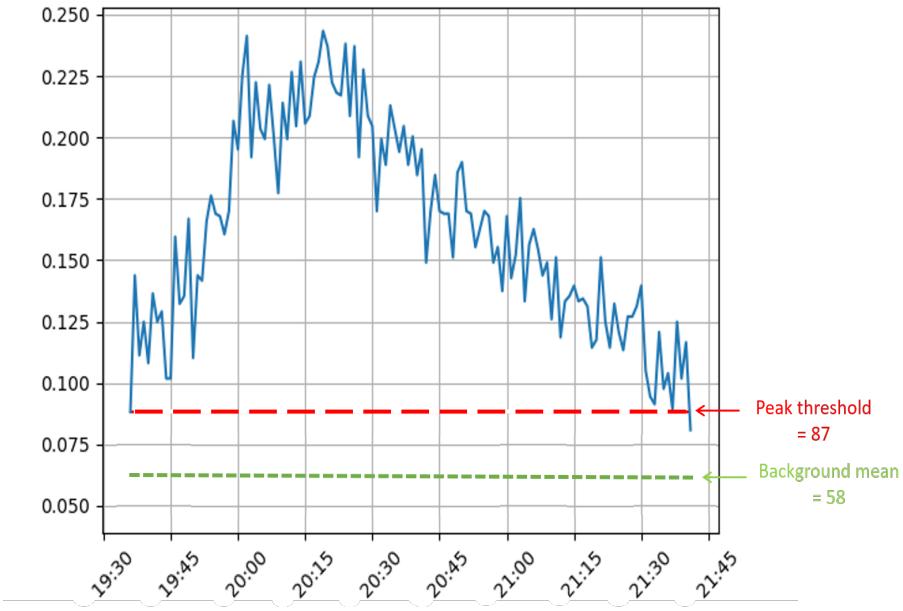

Figure 3.8: Example of an actual extracted incident

In the beginning, we considered all subsequences of the time series where the gamma dose rate went above the threshold as incidents. However, after experimentation, we found that short incidents were adding much noise to the data set and the clustering couldn't achieve any satisfactory results which is why we re-consulted the experts. After discussing it with the experts we found that they were most bothered by the *innocent* events that caused the gamma dose rate to go up and took a rather long time to go back down. That is because that behavior is the one similar to the temporal data's behavior when affected by a real threat. On the other hand, instantaneous events are immediately discarded by them as they explained high gamma dose rate for a short period of time is not harmful and hence not alarming. They explained how while monitoring the data, they are not alarmed by high gamma dose rate before at least half an hour has passed. To concludes, in the remaining work, we discarded all incidents that did not last at least 30 minutes above the peak threshold.

1. **Missing Data Imputation:** As we explained previously, the gamma dose rate data is very well susceptible to the missing data problem. That is

because we are dealing with data coming from sensors, and these sensors are most probably going to malfunction at one time or another. Because the data we are dealing with is relatively huge and based on the intelligence obtained from experts, we decided to deal with the problem of having missing data by: (1) dropping the whole time-series data (one year worth of data) if the missing data is distributed in big patches throughout it, (2) dropping the extracted incident if it encounters a missing data point because this means that the probe is malfunctioning at the time and hence it cannot be trusted.

2. **Scale Standardization:** The extracted incidents resulting from the extraction approach are of varying scale and amplitudes. The gamma dose rate can reach unpredictable levels when affected by a radiation event, we cannot know the maximum and minimum values in order to perform min-max or decimal-scaling normalization. For this reason, we had to discard them. On the other hand, **z-normalization** is highly applied in the time series literature. Its strong point is that it normalizes the samples, so only the *shape* of them is left to compare to each other. A value a of A is standardized to a' by computing:
$$a' = \frac{a - \mu(A)}{\sigma(A)}$$
The fact that it normalized the data to be of mean $= 0$ and standard deviation between 1 and -1 has great advantages explained in the next section.

3. **Length Standardization** As mentioned before in the state of the art, the elastic measure DTW is very sensitive to outliers, which means that if the variation in length between samples is too high, the clustering is not performed well as we will see later in the evaluation. To solve the varying length problem, a **Padding** technique has been used as proposed by Tan et al. [43]. Samples are padded with in-consequential data points such as zero or the mean or the median depending on the data distribution. By padding with zero to the z-normalized data, neither the mean (0) nor the standard deviation was affected since zero is indeed in-consequential for this distribution of data. Notice that without the z-normalization, it would have been impossible to apply the z-padding.
Thus resulting in having all the incidents in the dataset of equal length and without interfering in the characteristics of the data.
Preprocessing to a standardized length, the model was indeed able to produce the best results. The standardization applied in the preprocessing phase was critical for the approach. Without this preprocessing phase, the padding could not have been done and the other experiments were not giving meaningful clusters.

3.2.2 . The time-series clustering model

The main problem we faced while building our model is the varying length of the samples. As mentioned in the literature, such a problem has been properly addressed no where in the literature of time series clustering. There are algorithms that cannot work at all with samples of different length and others that "tolerate" the length difference. In order to decide on the best model we had to explore all the different algorithms to decide the best combination that suits our data.
With the big number of possible combinations we had to do a lot of experiments before admitting that none of the approaches gave us the results we were hoping for and deciding to propose a new improved model.
As explained before, one has to combine a similarity measure, which will calculate the distance between each pair of samples, and a clustering method which will use the obtained distances from the similarity measure to generate the clusters.

1. **Choosing the Similarity Measure:** Among the two elastic measures, we chose DTW as it tolerates slight time axis misalignment. Moreover, DTW is tolerant to samples of varying lengths. The same can be said about LCSS. However, between the two similarity measures, we found that DTW performed better with our data set than LCSS as the latter is more likely to ignore significant data points in the time series considering them as outliers. You will see in our experiments that our samples are basically made of outliers as they are abnormal behavior of the gamma dose rate, showing up in a stochastic behavior.

2. **Choosing the Clustering Algorithm:** Due to the preprocessing of the data with z-normalization and zero-padding, we decide to choose the **K-means with DTW Barycenter Averaging** algorithm for the clustering. Even if in the state of the art K-medoids is the most popular technique to be used with DTW, we will see that in our context K-means performs better as with the zero-padding samples become of equal length. DBA standing for DTW barycenter averaging [35], evaluates the mean of a set of sequences by iteratively refining the potential average sequence to reach the minimum DTW distance between it and the sequences.

3. **Choosing Optimal Number of Clusters:** Now that we have our clustering model, we have to choose the optimal number of clusters k which is the maximum number of clusters with no redundancy. In order to do this we had to experiment with different ks and evaluate the results of each. We first tried to do this using the indices mentioned in the previous section for determining the optimal number of clusters. We believed that if the data is well clustered, the indices will be able to predict the optimal number of clusters and in that case we will be able to do this autonomously.
   However, the results obtained from the algorithms were not helpful and sometime not logical. Even though, at the end, the silhouette method

predicted the right optimal number of clusters, we couldn't properly evaluate the method as we couldn't test it on a different data set that has another number of clusters. For this reason, we decided to evaluate the optimal number of clusters using the experts' help and leave automating this step to be considered in future work with more data sets.
In our approach, we presented to the experts the computed cluster centers from our experiments for $1 < k < 10$, and, together, we saw that for $k > 3$ we started to have redundant clusters (as shown later in Figure 3.18 for $k = 4$), so we decided that the optimal k for this dataset is 3.

3.2.3 . Evaluation

In order to compare the different approaches of the state of the state, as well as to see the benefit of our proposed model, we decide to evaluate in a systematic way different combinations of preprocessing phases (with or without z-normalization or zero-padded) with different clustering models (K-means, K-medoids, K-shape) as synthesized in Table 4.1. Several number of experiments was done as described in Table 4.1.

Clustering Algorithm	**Similarity Measure**	**Z-normalized**	**Zero-padded**	
			Yes	No
K-means	DTW	Yes	✓	✓
		No	✓	✓
	LCSS	Yes	✓	✓
		No	✓	✓
	Euclidean	Yes	✓	
K-medoids	DTW	Yes	✓	✓
		No	✓	✓
	DTW with length factor	Yes		✓
		No		✓
K-shape	SBD	Yes	✓	

Table 3.2: Model Experiments

3.2.4 . Experimentation

As explained previously, we had to experiment with all kinds of combinations between the different distance measure and the clustering algorithms discussed. In addition to that, we had to re-do the experiments with the modifications that we were proposing. Some of the different combinations we experimented with can be summarized in table 4.1.

The overall number of experiments done was 24 including the 16 described experiments in the table. The extra experiments are the different ones we did

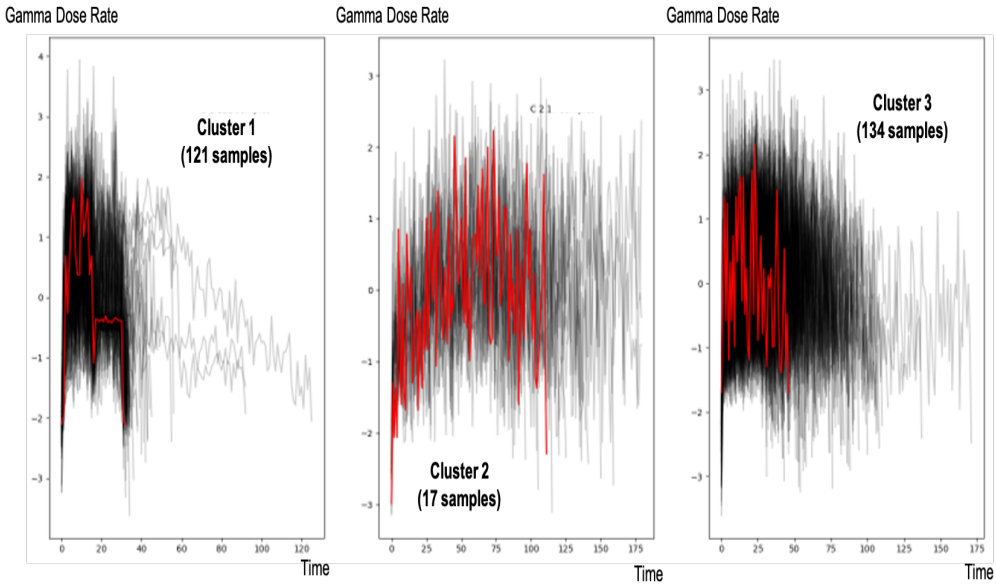

Figure 3.9: K-medoids with DTW and z-normalized data

using the hierarchical method as well as the ones were we used k-means with DTW without the DBA mean computation which was before we came upon the paper describing it and the shortcoming of k-means with this similarity measure .

Most of the reasons for discarding a similarity measure, clustering algorithm, or a suggested modification were explained previously in our approach. However, to elaborate our choices more we are going to present some of the cluster results we got using the other models before presenting the results of our approach's model.

- **K-medoids**

  Looking at the figures 3.9 and 3.10, using k-medoids with dtw with/without padding we faced the same problem caused by the fact that k-medoids algorithm tolerates outliers, so the obtained clusters have a lot of misplaced incidents and the centroid of the clusters does not clearly represent the observations in the cluster.

- **Same length algorithms**

  After applying the padding concept we reconsidered algorithms that only accept sample of equal length including the k-shape algorithm and euclidean distance. However, the results shown in figures 3.11 and 3.12 show how k-shape failed miserably while the other approach still didn't give the clearest clusters and most expressive cluster centers.

- **K-means**

  On the other hand, observing the results of k-means clustering, we can see how adding up each preprocessing step got us closer to the best cluster

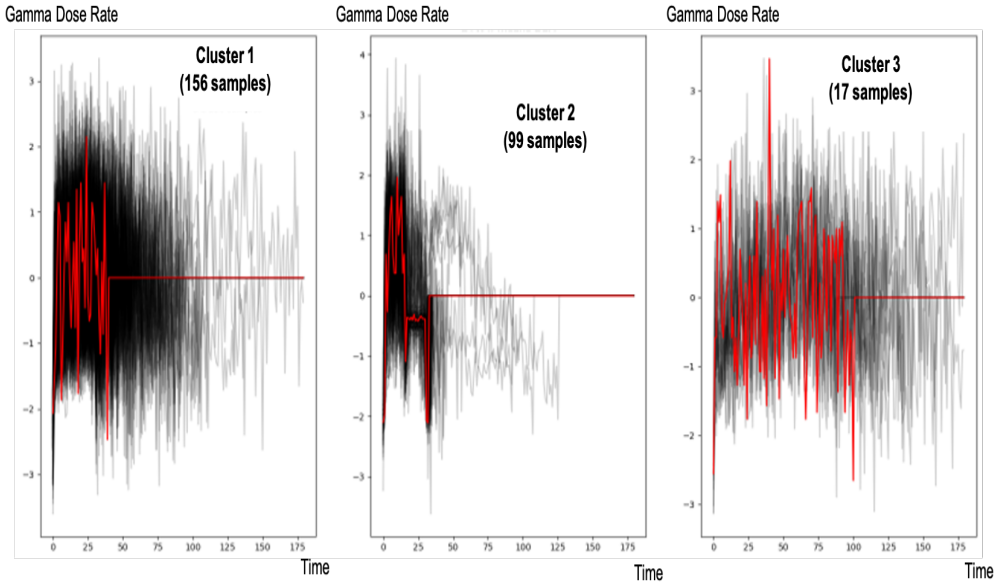

Figure 3.10: K-medoids with DTW and z-normalized data with padding

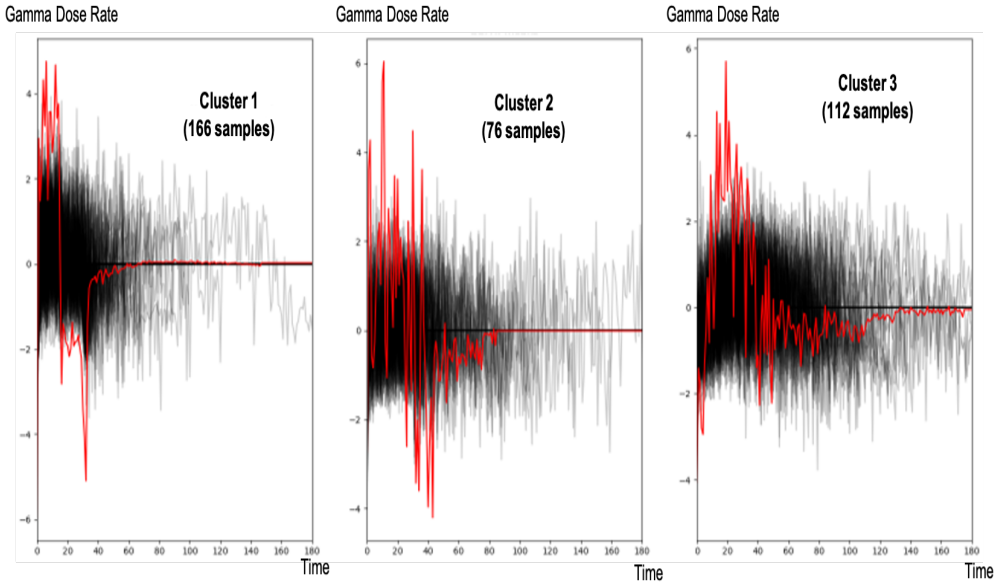

Figure 3.11: K-shape

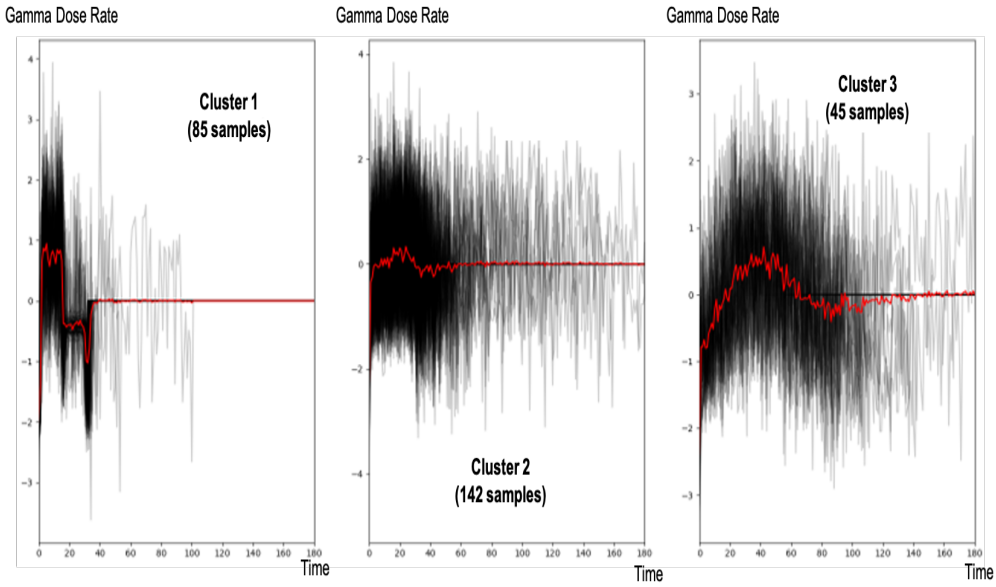

Figure 3.12: K-means with Euclidean distance

results shown in Figure 3.15 which were evaluated and approved by the experts. In Figure 3.14, the deformity of the cluster centers can be seen.

3.2.5 . Results Analysis

After evaluating the clusters obtained from our proposed model, we found that indeed the incidents in each cluster can be explained by a different underlying event. Each group has to be appraised in this phase to evaluate its "correctness" and identify the underlying event behind each. For this, a big number of each cluster's incidents have to be inspected to identify the underlying event and check whether all the inspected incidents of the same cluster have indeed the same underlying event and hence identify this event as the one behind the corresponding cluster.

For the results of $k = 3$ we inspected together, using the experts visualization tools, a big number of the incidents from each cluster using their different weather data (rain, wind, temperature...) to check whether there was a weather-related event happening at that time as shown in the figures 3.16 and 3.17, and if not using, their expertise and memory, we were able to understand other underlying events. In the end, we concluded that indeed the incidents grouped in each cluster were caused by a similar event.

In the end, we concluded that indeed the incidents grouped in each cluster were caused by a similar event.

- Cluster 1 's incidents are caused by a **calibration event** done on the probes.
- Cluster 2 's incidents are caused by a **stormy** rain where the wind causes

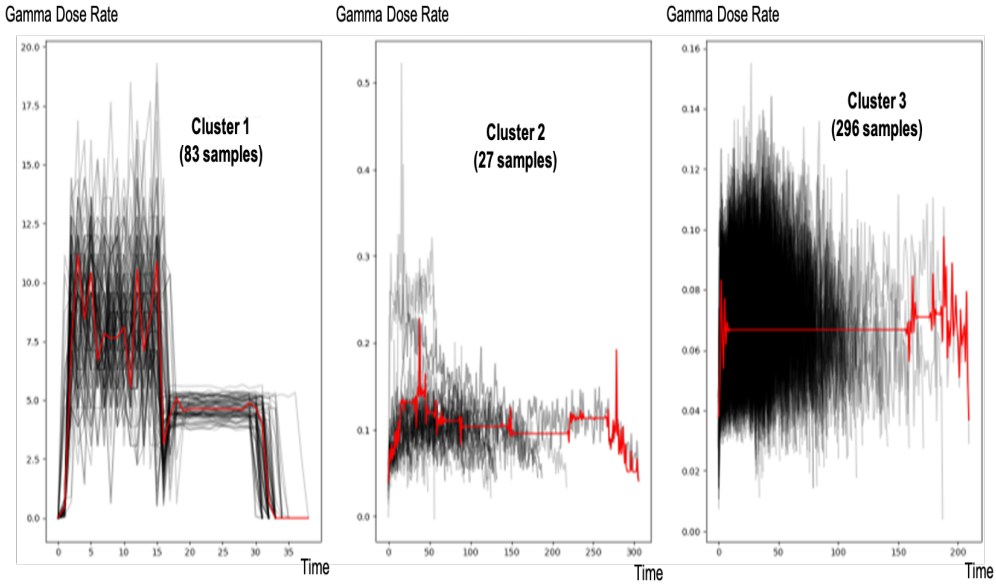

Figure 3.13: K-means with DTW and raw data

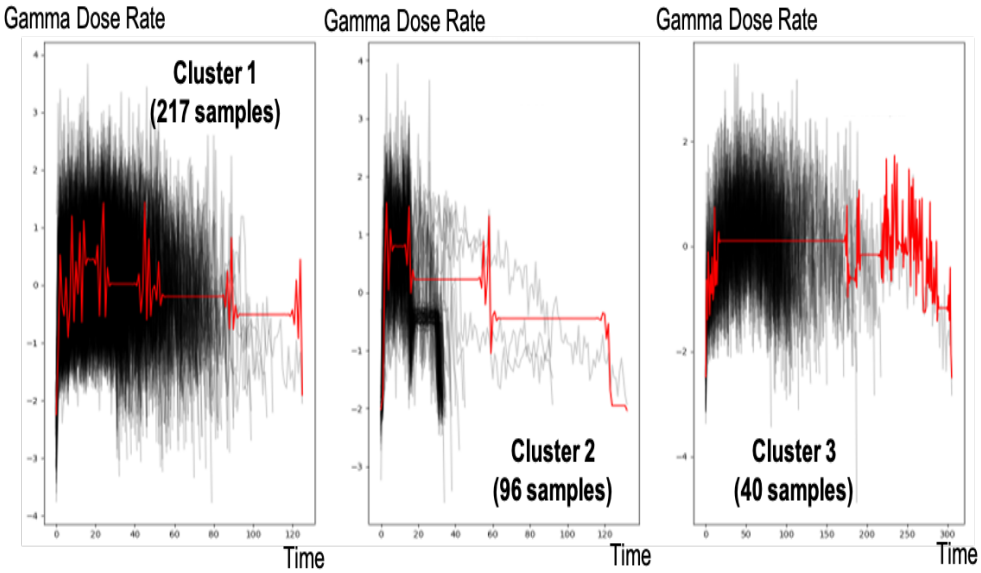

Figure 3.14: K-means with DTW and z-normalized data

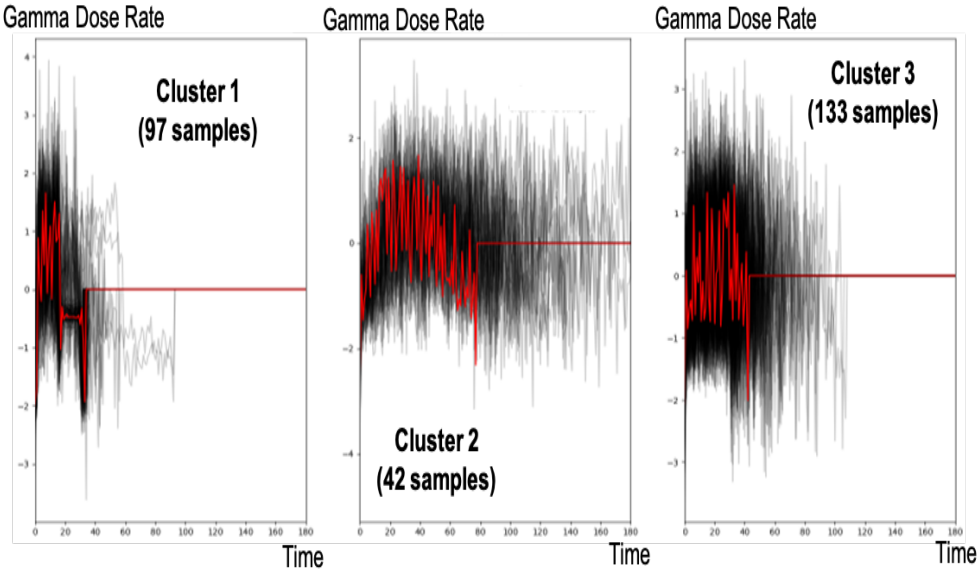

Figure 3.15: K-means with DTW and z-normalized data with padding

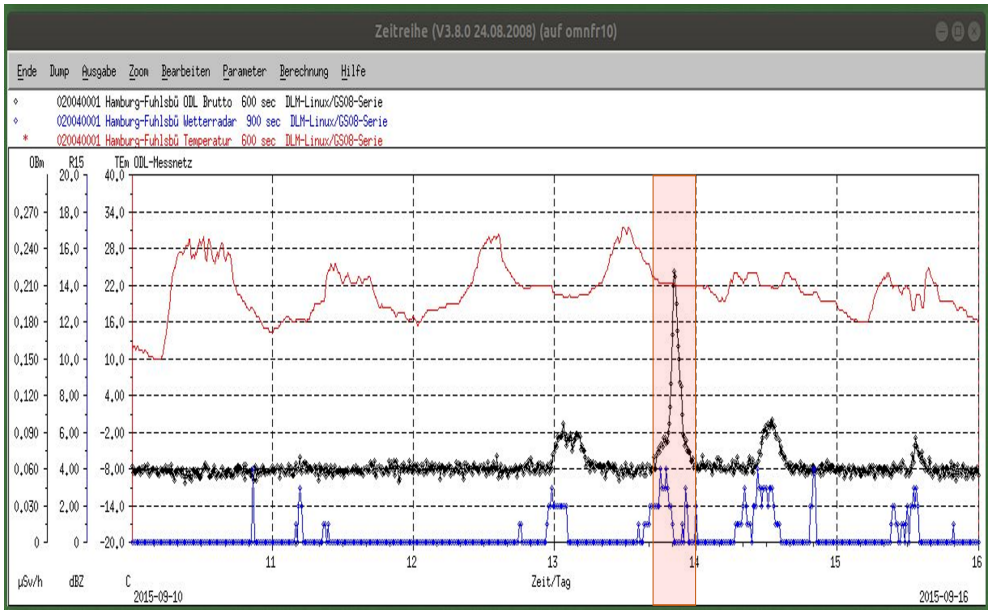

Figure 3.16: Precipitation, temperature, and gamma dose rate data at the time of an incident (highlighted in red)

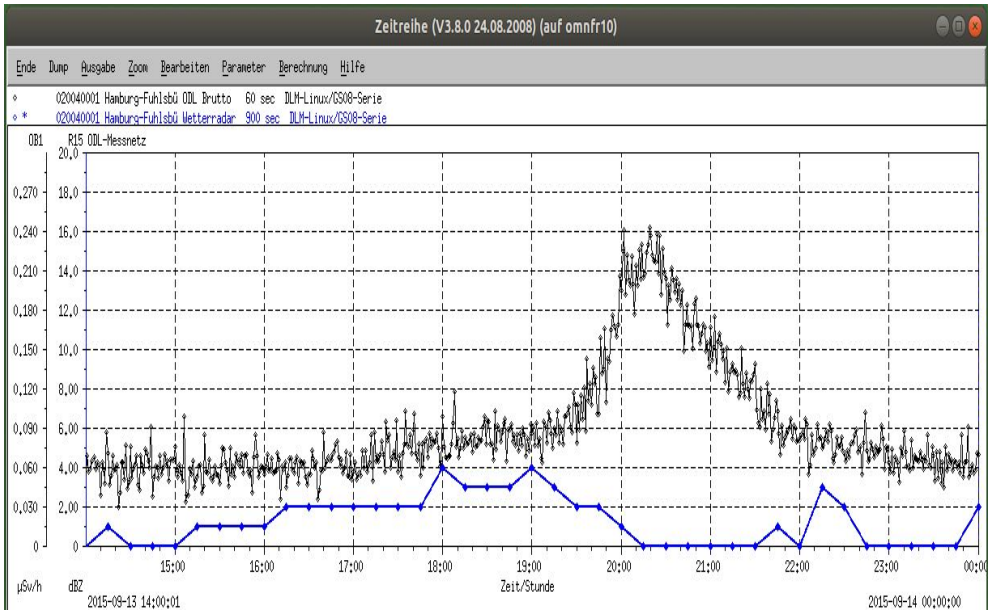

Figure 3.17: Precipitation and gamma dose rate data at the time of an incident (close-up)

the radiation elements not to fall at once into the soil but gradually causing the gamma dose rate to go up gradually as well.

- Cluster 3 's incidents are caused by a **normal rain** that causes the elements to go straight down and affect the probe with an immediate sharp increase.

Unlike the clusters of $k > 3$ where we started to see redundant clusters, as shown in the Figure 3.18 Clusters 3 and 4, based on our evaluation process.

3.3 . Conclusion

In this chapter, we presented our unsupervised machine learning based framework for autonomously identifying underlying events behind high gamma dose rate historical incidents. Our approach is to, after extracting and preprocessing the extracted incidents, apply a machine learning model that will group similarly behaving incidents as caused by the same underlying event using unsupervised clustering. The groups are then evaluated by the experts to recognize the events and, hence, label the incidents.

The proposed framework is divided into three different phases: Incident Extraction and Preprocessing, Time Series Clustering, and Cluster Evaluation. The first phase we did as a first step to prepare the data set for the machine learning model. In the next step we performed the two phases of clustering and evaluating.

In the clustering phase, we, specifically, tackled the problem of clustering time series of varying length which was properly addressed no where in the literature. We

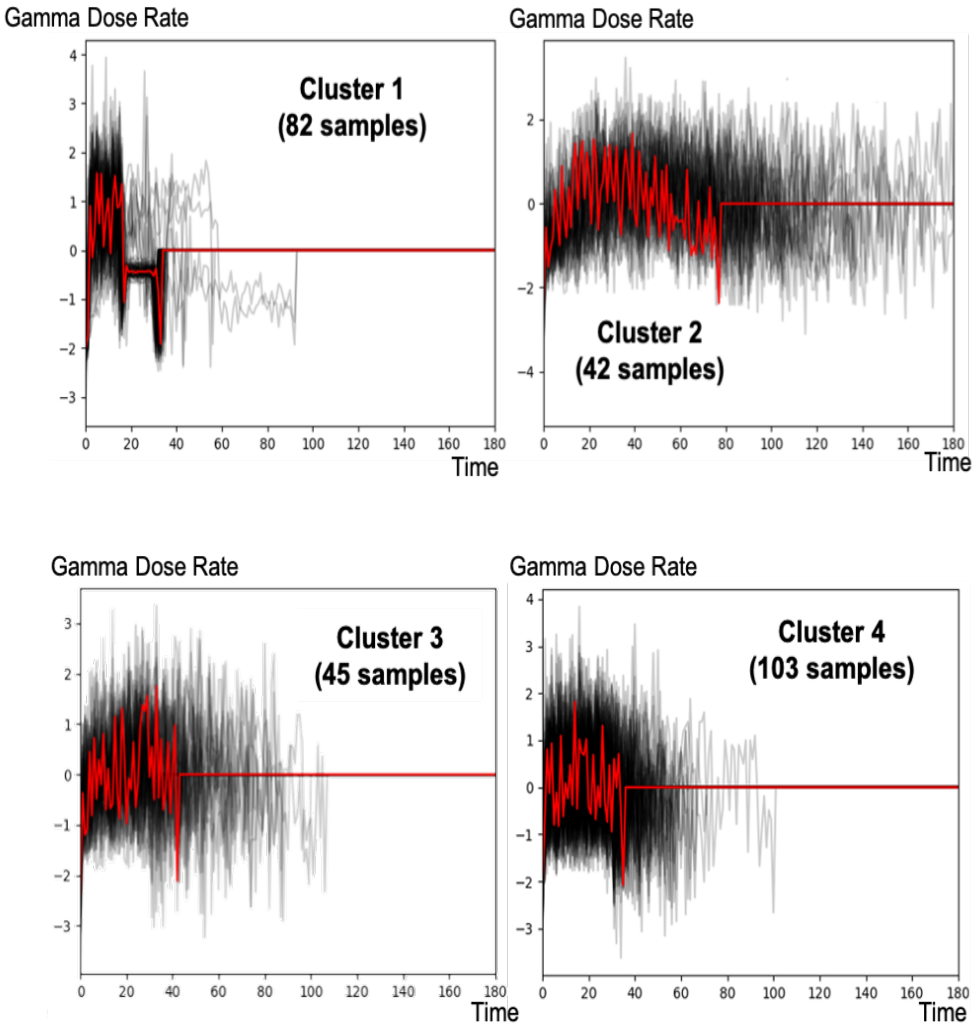

Figure 3.18: Our model's cluster results for $k = 4$

researched and experimented with the different classic and state of art approaches to evaluate their compatibility. When those approaches failed to properly cluster our data we proposed our model for Clustering Sequences of Varying Length.

Displaying the obtained clusters through the experimentation of various models, we were able to highlight how our model gave the best results comparatively. Also, the experts expressed positive and hopeful thoughts inspecting the results which motivates us to publish this contribution in an article and continue the work towards the intelligent REWS.

4 - Online Prediction for RIMI Framework

Contents

In the previous chapter, we were able to provide a detailed description of our approach for the Building the Predictive Model phase which is the first phase of the RIMI framework starting with incident extraction and passing through data preprocessing before reaching the clustering phase that provides several classes where each class groups similar incidents that are formed by the same cause.

In this chapter, we will focus on the Online Prediction phase which is the second phase of the RIMI framework. Since the classes were obtained in the previous phase, the main purpose in the second phase was to classify the online incoming incidents. For that reason, the time series classification methods were triggered in order to test their effectiveness in classifying the incoming incidents into their relative classes.

After investigating different approaches from the time series classification algorithms state of the art, we noticed that there is no specific approach that can result in a beneficial classification of the incoming incidents. So, we decided to move forward and to present a new approach for classifying the incoming discovered incidents.

Decidedly, in the current REWS, the manual intervention at every phase of the analysis process is heavily time consuming, labor intensive, and risk prone. Thus, an end-to-end automated system is a critical need to address the shortcomings of REWS systems.

In the previous chapter, we discussed the first phase of the RIMI framework. We mainly focused on the design of the first phase, introduced our contribution, and explained the workflow of the initial version of the solution. The two main phases of the proposed framework are: building the predictive model and (near) real-time detection and prediction. In the first phase, the historical data were analyzed to extract knowledge about the previous incidents occurred in the past. The evolving parameters problem were solved by preparing a catalog of parameters before the extraction process starts. The evolving issue was tackled by a specific algorithm through several steps of calculation to make sure that the parameters obtained by the end of each month are accurate. Then, the extracted incidents undergo a special preprocessing phase to make sure that are ready to enter the proposed clustering model. This was done using a z-normalization preprocessing step that was responsible for dealing with the scale issue since the extracted incidents were not at the same scale. Another preprocessing step was applied to deal with the different length incidents issue. This issue was tackeled by applying the zero padding method on the extracted incidents. Eventually, a huge group of incidents were obtained and prepared to be clustered through specific clustering model. Moreover, the clustering model introduced was formed by combining the similarity measure (DTW) [19] as well as the clustering algorithm (K-means) [28] with its averaging method (DBA). This model resulted in forming several classes

of incidents based on their geometrical representations and cause annotation.

For the Online Prediction phase, it was clear for us that the next step will be matching the incoming incidents to those previously clustered from the first phase. This will result in identifying the incoming incidents as soon as possible and give the correct impression to the experts so that they can distinguish the innocent incidents from those that are critical. After tackling all the shortcomings behind the first phase of the RIMI framework, we reached the step where we need to start our investigations for the second phase and hence propose our contribution for the Online Prediction phase.

At Prediction phase, the incoming readings are analyzed to explore the thresholds in real-time. Once an incident is detected, the data preprocessing model is used to deal with the scale and length issues and a matching process is performed to search for similar incidents in the predefined classes. This helps to identify the real cause of the current incident.

Since the first phase results in several classes grouping similar incidents, the second phase goal was to match these incidents to those occurring at real time. Although the contribution done in the first phase was based on clustering algorithms due to the unlabeled extracted incidents, it was clear for us that the second phase should depend on classification algorithms since the matching process is trying to classify the incoming incidents within labeled classes formed at the first phase.

4.1 . Time Series Classification

Time series classification is a late bloomer in the domain of machine learning. The significant issue for this type of machine learning is that it gives importance for the order that the data is coming by. Classification is the process that can be applied over structured or unstructured data [46], to train a model aiming to have the ability for this model to predict new unseen records in the future. So, it is the process of categorizing unlabeled data into a number of given classes/categories.

4.1.1 . Classification Types

In statistical data analysis, two major approaches are represented. The first is Univariate data analysis with single variable of interest. In this case, variable is influenced by a single factor. The second approach is Multivariate data analysis with two or more variables taken as input. In such case, variable is affected by multiple factors.

- **Univariate Time Series Classification:** It represents a sequence of readings or measurement for a single variable. Over time, one variable is a point of interest to be monitored and recorded [47]. The obtained data describing a certain event or phenomena are observations over time for fixed intervals.

This Univariate data can be represented by calculating estimations for each period of time such as finding the mean, variance, standard deviation, and others. Machine learning algorithms can easily be applied on Univariate Time Series Datasets. They are simple and can easily be understood. There exist a number of tools that are used to plot Univariate Time Series Data. The x-axis represents the time intervals and the y-axis is dedicated for the measured value at that point of time. Usually, line graph is used to plot Univariate Time Series Data. In Univariate Time Series Data there is no concern about variables correlation because there is only a single variable observed during time progress.

- **Multivariate Time Series Classification:** It is considered as extension of Univariate Time Series Data. Two or more variables are described at every point in time. For a specific period of time, and at each fixed interval within this period, each variable is represented by a value. Information gained from multivariate Time Series Data is the result of analyzing all of the variables at that time [48]. Relations between variables could be dependent or independent. This relation is analyzed by finding the correlation between those variables. Correlation if negative, indicates that whenever one variable increases the other variable decreases or vise versa. Positive correlation between two variables is when both have the same behavior over time. In some cases variables could be independent of each other, here correlation will be zero. Correlation coefficient can be described as strong correlation when the value is between $\pm$ 0.50 and $\pm$ 1, medium correlation when value between $\pm$ 0.30 and $\pm$ 0.49, and small correlation for value less than $\pm$ 0.29. Therefore, multivariate Time Series Dataset is more challenging than Univariate. Great results are obtained when applying machine learning algorithms with multivariate Time Series Datasets.

4.1.2 . Important terminologies in classification (Definitions)

- **Classifier:** represents the algorithm that decide how and why to map unlabeled data records to a particular class. Each classifier has its own number of parameters for best tuning. Those parameters represent how the classifier should treat the input data to learn from. It is very important to be careful and accurate in specifying those parameters. Furthermore, they are highly tied to the nature of the data that will deal with. So, by understanding the data, setting classifier parameters will be much easier and thus leads to obtain a well-trained model.

- **Learners:** are divided into two types, lazy learners and eager learners.

  - **Lazy Learners:** The way Lazy Learner works, is that it stores the training data until the testing data arrives [49]. Then, classification

operates based on observing the most related stored training data. Lazy learner has less training time cost but high predicting time cost. An example is the K-nearest neighbor classification algorithm.

- **Eager Learners:** The Eager learner classification algorithm starts by constructing the model from the training data before classifying new/unseen data [50]. For high performance, the model should reach a single hypothesis that covers the entire instance space. Therefore, model construction requires more time in training but less in predicting. Example is the Decision tree classification algorithm.

- **Training/Testing data sets:** a set of labeled data that will be used to train the model over. In the beginning the selected classifier and after setting its dedicated parameters it should be trained by running it over the selected training data set that is well prepared for that purpose. The more the training data set is clear and rich the better the model will be trained thus resulting in high accuracy when tested by running it over the testing data set.

- **Feature:** It is a single property of an event, object or phenomena that is measurable and has a clear value. Some features are related to time making its order an important factor, others are independent of time and ordering will have no significance on the trained model.

- **Classification model:** Once the classifier is selected along with its parameters, now it should be trained by preparing as an input the training labeled data set, which will result in having a trained classification model that will be tested by running it over testing unlabeled data set to see its accuracy in predicting classes.

- **Binary Classification:** When the expected output falls in either one of two possible outcomes. As an example, the case of classifying emails as spam or not spam. In this situation each email will be assigned to either spam class or not spam class, no other possibilities.

- **Multi-Class Classification:** where the output of the classification model can be one and only one possible class from a list of candidates. The number of possible classes is fixed and greater than two. Observations are labeled with a single class. As an example, training a model to classify the weather by predicting one of the pre-defined classes (greater than two) such as Rainy, Stormy Rain, Sunny or Cloudy, and thus each observation will be assigned to a particular class.

- **Multi-Label Classification:** where the expected output of classification process can be under more than one class. In other words, a single record

or observation can be assigned to multi categories. For example, classifying movies which can fall into one or more of pre-defined categories such as, Horror, Comedy, Thriller, Action, or Romance, and here each movie can have multi labels (ex: Action-Comedy Movies).

- **Model Accuracy:** After splitting data set into Training and Testing sets, and once the model is trained over the training data set, now it is the time to evaluate the model by running it over the testing data set after omitting the label column. For the model, those are new records, and it should predict the class for each. Then, the predicted classes are compared with the real classes to test the performance of the model. A number of methods are used to evaluate the classification model. The easiest way in testing the performance is to divide the number of correctly predicted over the total number of predictions.

4.2 . Building an Online Classification Model

Investigating the classification algorithms that are presented in the literature, we noticed several shortcomings that prevented us from relying on a specific algorithm for our online classification model. After passing among the classification algorithms introduced in the literature, we noticed that there is no specific algorithm that can fit our data perfectly since our data has special characteristics and behaviour. We noticed that although some algorithms work perfectly for a specific type of incidents, they were unable to succeed classifying other types. This problem was enough for us to not trust a specific classification algorithm when dealing with our incoming incidents. Moreover, we noticed that these algorithms were unable in detecting incoming incidents that are of unique behaviour and should be classified in a new class that will be labeled by the experts later.

Several work in the literature in the different research domains of financial, retail, aeronautics, and many other domains are done on specific data sets that have certain characteristics because of being done in an experimental environment for research purposes. However, what we want to achieve in this work is to extract intelligence from the raw historical data. Thus, we proposed a machine learning based framework to automate the event identification process so as to decrease the time and efforts spent as well as to increase the efficiency and accuracy of the process.

What we are hoping to reach in this thesis is to, as autonomously as possible, match the incoming incidents to those clustered in different classes that were formed in the first phase. This will help in identifying the cause behind the incoming incident as soon as possible and will provide the experts with a wide knowledge about the current situation occurring near the probe related to the incoming incident.

As we aim to match unlabeled incidents without any human intervention as

soon as possible, our research is in the field of the supervised machine learning time series classification.

Our proposed framework is divided into four phases which is briefly explained as follows:

- **Incident Identification:** alarming incidents exceeding the peak threshold value and lasting for at least 30 minutes above the peak value defined.
- **Data Preprocessing:** the raw incoming incidents will undergo the proper preprocessing so as to have a data set ready to be matched with previous incidents clustered at the first phase.
- **Time Series Classification:** the machine learning model is designed to match the incidents into those presented in different groups (clusters) formed by the previous phase.
- **Classification Model Evaluation:** the results of the classification model are evaluated with the help of experts to determine whether the incoming incidents are matched correctly to those previously clustered and to check if the cause behind the incoming incidents was guessed correctly.

A main contribution in our work is researching the techniques used specifically when classifying univariate time series data. The existing classification algorithms presented in the literature were not able to return the desired target behind the second phase if we rely on one of them to perform the task, which had us introducing our approach in combining these algorithms and choose the best outcome from each one of them upon facing an incoming incident.

The main research question of this phase hence could be formulated as follows:

"What is the machine learning model that should be used for online time series classification of special behavior and how do the different models perform in practice ?"

In that direction, we present a comparative study between the different (time series) classification techniques that are available. We compare these techniques based on applicability and effectiveness. Besides conducting a literature study, we also apply the different techniques on our data set to test their performance, and when none of the approaches give satisfactory results, we will propose a new improved model which gave the best results through our testing.

Data analysis helps in deriving new conclusions and supports decision making. Many approaches are exhibited in the field of data analysis. Each approach introduced its own technique and methods under different names to describe how it handled data analysis. Besides, many tools are used to fulfill this process with best practice especially when it comes to Big Data issues. Today, science and nearly all businesses rely on data analysis for extracting knowledge and conclusions, and supporting them in decision making for better operation in the future.

4.2.1 . Generic Classification Methodology

The classification model should tackle different classification issues. Thus, several steps, starting from data processing, passing by choosing the classifier, and ending with evaluation, should be followed to run the model. However, sometimes it is very hard to tell which classifier to be used and which one is superior to other. Selecting the appropriate algorithm depends on both the nature of the application and the data set.

1. **Data understanding and processing**

   Data understanding and processing is the cornerstone and the starting point in the process of building a classification model. It is described as the process of inspecting, visualizing, understanding, cleansing, transforming, and enriching raw data with the aim of discovering hidden information and insights found in it. Data understanding and processing is essential. Data can undergo many preprocessing phases to ensure the best quality of data. Removing outliers and unnecessary data will reduce model confusion and enhance its overall performance. Other techniques such as, adding valuable features to the data, data normalization, and padding causes the model to significantly learn from this data and thus, increases model accuracy in prediction. Therefore, the higher the quality the data is the better performance the model will give.

2. **Classifier Selection and Initialization**

   Once data is analyzed and understood, now it is clearer on selecting the optimal algorithm that will best fit the data. So, it is not just about try and see what the result will be, then select the best model algorithm. Therefore, based on the data study done earlier, one can narrow the selection of possible classifiers that could fit the proposed problem. A large number of classifiers exit in field of domain, and each tackles in a different way the problem found in the data. For any selected classifier, the procedure is the nearly the same. So, after selecting the intended classifier which must fit well with the data, now comes the need to dive into all possible parameters that should be predefined for the selected algorithm to function based on it, thus causing it to run in an efficient manner. Each parameter has its significance on the overall performance of the model. Some are critical for example, selecting the distance based measure technique for classifiers such as K-nearest neighbors. Other parameters are less important and usually are left to its default values (But even though still in many cases you need to alter those parameters to reach the intended results) for example, the number of iterations the model should perform to learn from the data. Therefore, nothing is arbitrary, all should be best selected, and once parameters are defined now you can instantiate the classifier object and ready for the next phase, that is training the model with the training data set.

3. **Splitting Dataset between Training and Testing**

   Classifiers require the existence of sufficient data set for prediction. If the data set is small, it will so difficult to split it between training and testing data sets. Therefore, from the beginning data set must be sufficient enough to perform the split. This number of observation within the data set could reach thousands or even millions. The split is not done arbitrary or randomly. Both data sets training and testing should represent very well the problem domain. Not to fall in the problem of unbalanced data specially when it comes to the training data set. Unbalanced data set is when the distribution of classes within the data set is unfair, which means the occurrence of a particular class is greater than others. Therefore, to obtain higher performance of classification models, data set should be balanced. In case limited data set, a very popular technique is used to full fill this gap. This technique is the k-fold cross-validation [51]. This technique divides the dataset into k groups. During the training phase, it holds one group for testing and trains the model with the rest. In this way the model will be learning from all of the observations in a sufficient and effective way. There is no optimal spit for data sets. No one can generalize the percentage of training and testing data sets. All depends on the problem domain and the objective of the model that is going to be used. The only thing that all agrees on is that the training data set must be greater than or equal to the testing data set. As an example for most common data set split:

   - Training 50% and Testing 50%
   - Training 67% and Testing 33%
   - Training 80% and Testing 20%

   Therefore, two main rules for data sets, the first is to have a sufficient data set for the model to learn and the second is be sure that the data sets are balanced and represents well the problem domain.

   - **Training Phase**

     The classification model accepts the training data set as input. Then it starts to iterate through all the observations in this date set. Each observation is of two parts the input values and its corresponding output – the label –. The model learns by mapping the input with the output and tries to conclude with a set of rules that will be used later on for prediction. This iteration the model performs is called model fitting and usually the method used is fit(X, y) where "X" is the input data set and "y" represents the output for each input.

   - **Testing Phase**

The classification model is now trained but not tested yet to calculate its performance and accuracy in prediction. The testing data set is held for that purpose. The testing data set contains a number of observations that is less than or maybe in some cases but rare equal to the number of training data set. The difference is that in testing data set the label column containing the outputs/classes is omitted leaving it for the trained model to predict. In this way, the model will be tested and after that a verification process will be executed to see how accurate the model is and if it predicted correctly the classes of given observations.

4.2.2 . State of the Art

In this section, we briefly recall the main Time Series Classification Algorithms mentioned in the literature. We compare these techniques based on applicability and effectiveness. Besides conducting a literature study, we also apply these different techniques to our data set to test their performance.

For time series data, there exist several algorithms that consider the time factor, which is essential in our study. A common problematic solution that could happen when dealing with time series data is to treat each value in the sequence as a separate feature. This is the core difference between time series data and Tabular Data. In time series data, the order of the data is essential and critical. In contrast, in Tabular Data, the order is ignored and scrambling the order of the features will not affect the prediction process. Therefore, each algorithm dedicated for time series data is based on a technique and perspective that extracts knowledge from the time series data concerning the order of the data.

Those algorithms are categorized as follows:

- **Distance based algorithms:** This type of algorithms represented in Figure 4.1 rely on distance metrics to find the optimal class membership. It plays an important role for pattern recognition problems. Selecting the similarity measure upon which the classifier will perform the identification of the class will affect the accuracy of the model, and the time and space complexity. Most popular distance measures used are Euclidean [18], Manhattan [] and Dynamic Time Warping with Barycenter Averaging (DBA)[35].

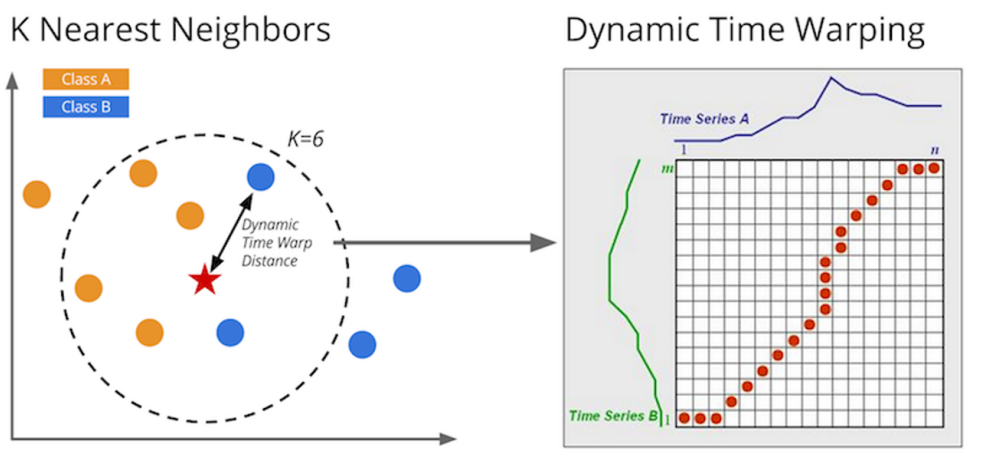

Figure 4.1: Distance Based Algorithm

K-nearest neighbor (KNN) with DBA: is the most popular and efficient distance measure classifier used when dealing with Time Series Data [52]. This type of algorithm is simple, easy to use, robust, and requires few hyper-parameters tuning. Since the selected similarity measure is the DBA the computation cost is high.

- **Interval based algorithms:** This algorithm depends in its classification on the information retrieved from various intervals of series. Time Series Forest classifier is a classification technique that is built for this type of algorithms [53]. TSF adopts the random forest classifier technique and apply on Time Series Data. The TSF classifier during classification, it splits the series into random intervals. Each interval has a random starting and ending point, which makes each interval of different length. Feature extraction is used to summarize each interval. Features can be the estimation of mean, standard deviation, and slop, and then each interval is represented as a single feature vector. Similar to the random forest algorithm, TSF constructs a number of decision trees from the extracted features. Once all decision trees are constructed the class with higher votes will be selected. Therefore, the classification depends on the majority of votes of all the trees in the forest as shown in the Figure 4.2.

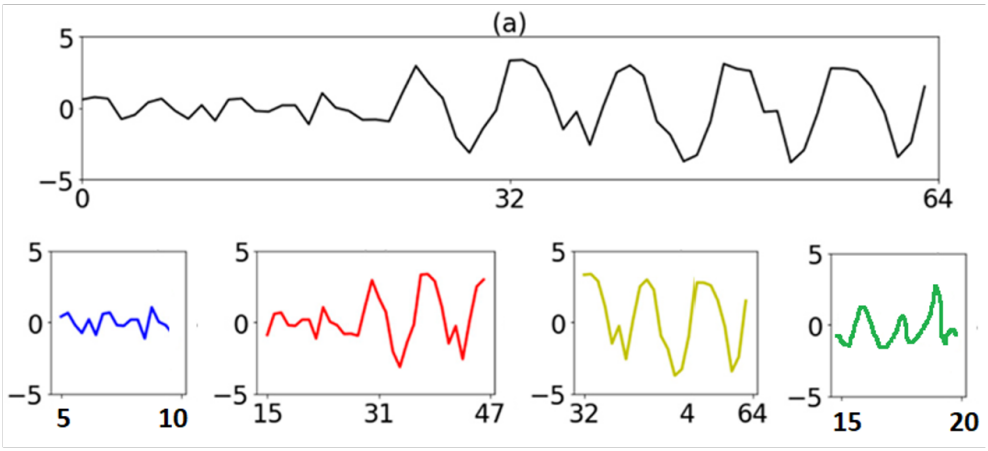

Figure 4.2: Interval Based Algorithm

- **Frequency based algorithms**

  Classifiers that follow this type of algorithm rely on frequency of the extracted features from the Time Series Data. Random Interval Spectral Ensemble known as RISE is a straightforward classifier that is similar to Time Series Forest [54]. Therefore, this algorithm constructs decision trees and upon the majority of votes the classification takes place. Two fundamental points RISE differs from TSF. The first is in the way series are split into sub intervals. RISE breaks the series into single time interval length for each tree, while in TSF as explained earlier a series are split into intervals of varying length. The other point in difference is in the type features that are extracted from the intervals of a series. RISE extracts spectral features from the series' intervals and trains its decision trees, unlike TSF which relies on summary statistics when performing feature extraction as shown in the Figure 4.3.

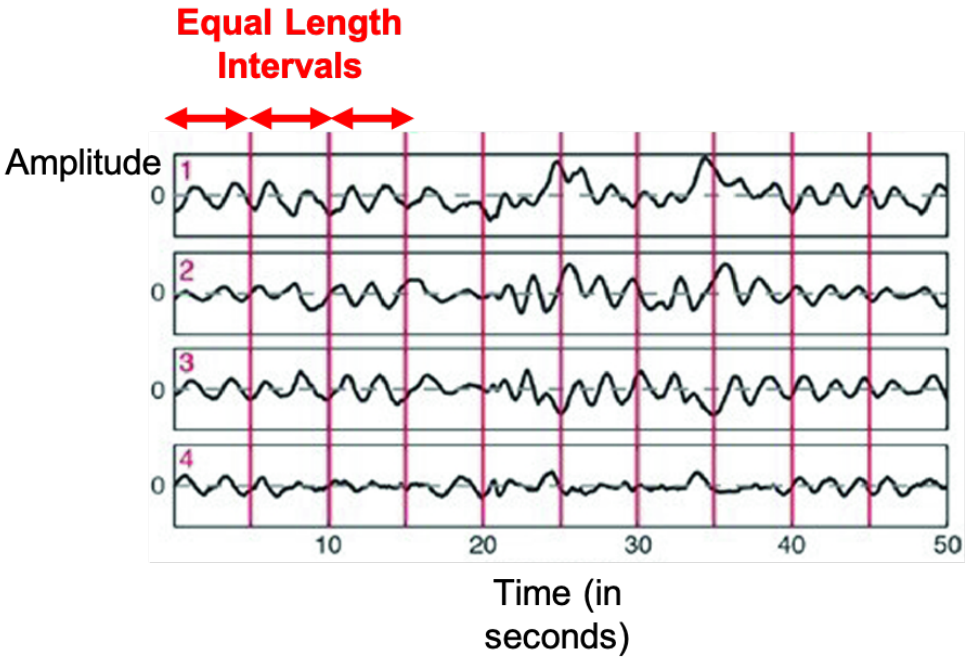

Figure 4.3: Frequency Based Algorithm

The spectral features that are extracted from the Time Series Data are series-to-series feature transformation, which means that the extracted features are not primitive values such as mean or standard deviation, instead it is another series, that's way it is called series-to-series feature extraction. The extracted features include fitted auto-regressive coefficients, estimated auto correlation coefficients, and power spectrum coefficients (Fourier transform coefficients). RISE is a straightforward classifier, the spectral feature extraction is a built in process that runs within the classifier. The only main parameter that should be defined is the number of estimators that is the number of decision trees to be constructed.

- **Shapelet based algorithms**

  Shapelets are sub-shapes or subsequences of time series. Shaplets are used as representative of a class [55]. The main objective of shapelet-based algorithm is to identify for a particular class the bag of shapelets with discriminatory power. Each shapelet is an interval extracted from a time series and it should follow the same order as shown in the Figure 4.4. Example, consider the following series [1,2,3,4], from this series only 5 possible intervals can be extracted, [1,2], [2,3], [3,4], [1,2,3], and [2,3,4]. Now the algorithm will select the most significant intervals that will distinguish this class from the others. This is what is meant by shapelets with discriminatory power.

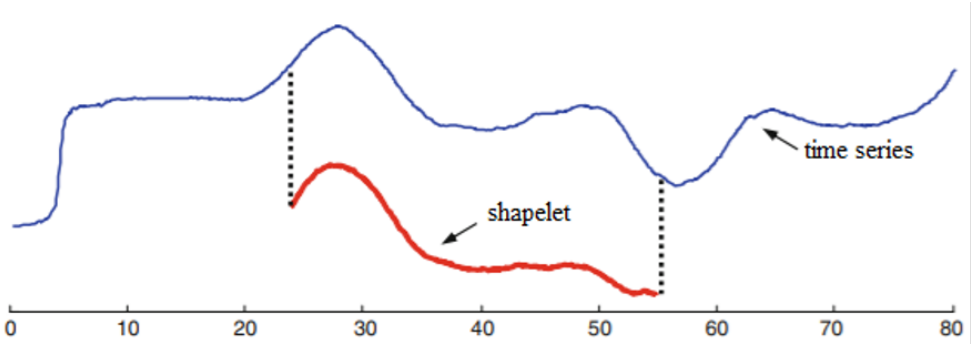

Figure 4.4: Shapelet Based Algorithm

During classification, the Shapelet based algorithm transforms the incoming data sets into "K" shaplets yet to be compared by the "K" shaplets extracted for each class in the training phase. The comparison is based on the distance the "K" shaplets. Any vector based classifier can be implemented to perform the classification process depending on the extracted shapelets.

In their paper [56], the authors introduced Time Series Data and Time Series Classification methods, focusing in their research on the importance of distance based classifiers, and although it is a simple method but it has a significant impact in the domain. The most common distance based classifiers are K-nearest neighbor (KNN) and Super Vector Machines (SVMs) [57].

Xing et al 2010 [58], in his paper divided Time Series Classification method into three main categories. A feature based methods, model based methods, and distance based methods. First, feature based methods, in such methods the Time Series Data are transformed in to feature vectors. Those feature vectors could be of primitive types such as mean, standard deviation, and slop, or spectral features were spectral methods are included. Those spectral methods are also called Series-to-Series feature extractors, for example Discrete Fourier Transform (DFT) [59], and Singular Value Decomposition (SVD) [60]. On the other hand, there is the model based methods. This type of methods considers that all Time Series Data related to a particular class are generated by the same underlying model. When new Time Series arrives it will be routed to the most appropriate class based on the model that most fit this new data. Models rely on special features found in the Time Series Data. Some models based approach uses auto-regression coefficient module in classifying unknown Time Series Data, example Bagnall and Janacek 2014 [61], and Corduas Picollo 2018 [62], or uses hidden Markov Model [63]. Finally, the Distance-based methods, which relies on the distance similarity measure, and it falls under two categories, the Lock-step Measures and Elastic Measures. The Elastic Measures is the most used when it comes to Time Series Data, because of the time variation. Euclidean is an example of Lock-step measure technique and DTW is an example of Elastic measure technique. Once the distance similarity

measure is identified it can simply be applied to any distance based classifier such as KNN and SVMs.

4.3 . Proposed Online Classification Approach for Gamma Dose Rate Incidents

Developing the overall approach was a chain of related continuous work that has been started earlier. Starting from the first phase where we presented our clustering model, many improvements and enhancement are applied to the extracted incidents and many questions have been answered in order to proceed with the next phase where the online classification approach is introduced to collect the profit behind the previous work done.

4.3.1 . Online Classification Approach

Online Classification is the process of identifying the class of new incoming incident, as soon as possible, based on the learning steps which are performed by the classifier by iterating through the training dataset. This training dataset is the output of the clustering phase where a new column is added to the data containing the class of each incident. Classification is a supervised machine learning technique which requires the existence of both dataset and label. Therefore, after clustering, each incident is labeled by a class. So incidents can be classified as rain, stormy or probe calibration. Moreover, there exists a number of time series data classification algorithms where each algorithm deals with the time series data from a certain perspective.

Incidents are generated from different probes of different locations. Each probe has its own settings and behavior in generating the data, since each has its own values of background means and peak threshold. Hence, normalizing the data is a must especially when the built model for classification is trained by normalized data. Thus, the classifier will perform better when new incident arrives. Other challenging point is the one related to the length of the incoming incident. The classification model is trained over full incidents, and when deploying it in online classification, the arrived incident is a part of incident that is not completed yet and the classifier should identify it as soon as possible. That's the general aim and objective behind the overall system. The system should be enhanced in way to identify the incident before it ends so that experts have enough time to alert the region where the probe has sensed the high gamma dose rate in case it is a real incident and not innocent. Therefore, the classification model should be able to detect the behavior of the incoming incident part and based upon this behavior a prediction to the incident cause should be done. Furthermore, going back to the classification techniques that are dedicated to solve similar classifying problems, there exists a number of classifiers each based on a hypothesis that characterize time series data from different perspective. Some are distance based and they

depend on the similarity measure to identify what is the most probable class of this new incident. Others are shapelet based classifiers, looking at the shape an incident is characterized by, where each incident is represented by a bag of shapelets that are sub parts of the original shape. Therefore, selecting the appropriate classifier technique was not that easy or a straightforward process to implement. Besides, it was not beneficial for us to have some incidents classified perfectly by an algorithm while others are better with other algorithm.

4.3.2 . Online Classification Model Framework

Our proposed approach can be divided in three parts where the output of one part is the input of the next. All of the three parts are mandatory. The first part is described as the preparation part. In this part the data generated by probes are grouped into incidents. By grouping we meant that incidents are formed of time ordered readings of gamma dose radiation that has lasted for 30 minutes or higher above the peak threshold. Those extracted incidents will then undergo some pre-processing phase techniques to obtain the desired results from the classification model. In the second part, the processed incidents will then enter the classification phase where classifier will be implemented to take incidents as input and thus predict its class. This part is called incidents identification/classification where unlabeled incidents will be assigned to a single class from the pre-defined classes. Finally, the last part is the evaluation part. In the evaluation part the predicted class of an incident will be verified to check whether the predicted class is true or not, and if it is possible that this incident is a new class that needs to be handled later on.

An automated machine learning framework will be introduced for online incident Time series data classification. Aiming to decrease the cases of false alarms that cause panic for experts, especially when raised due to innocent events such as rain that has triggered the alarm, and if the case was real a fast action should be considered.

Describing the three parts explained above, the proposed framework is brake down into four phases. Each phase will be explained in a brief way:

- **Incidents Extraction:** Only what is called alarming incidents are extracted from the data generated by the probes. And when we say alarming incidents we are talking about incidents that have reached and peak and remained for more than 30 minutes and thus, it must be identified.

- **Incidents Preprocessing:** Extracted incidents will enter a processing phase to be ready for analysis by the next phase that is the classification phase. Without this phase, the classification model performance will drop, since it has already been trained on a certain characteristics of data.

- **Time Series Classification:** A machine learning model is designed and implemented to accept the processed incidents from the previous phase as

input and then its role is to map this incident to what most likely class it should be assigned to.

- **Classification model Evaluation:** With the help of Experts in the domain we will evaluate the classified incidents to verify the performance and accuracy of the classification model for better enhancement in future predictions.

1. **Incidents Extraction**

   When extracting incidents online directly after probes has sent the readings, what only we care about are incidents that exceeded the peak threshold for more than 30 minutes, because if the duration is less than that or the gamma dose rate is lower than the pre-defined peak value the gamma dose radiation is not harmful, thus there is no need to investigate in the cause. That's why experts concern is only with incidents that lasted that period of time above the peak. Moreover, innocent alarming incidents are the ones that we are interested in identifying to reduce experts hard work and investigation in case it happened. Innocent incidents behavior is similar to the one caused by real threat, in term of causing the gamma dose rate to go up the peak threshold then starts to decrease over time and goes back down to normal.

   As we explained earlier, the classification model is designed and implemented then trained by the dataset that was the result of the first phase (clustering phase). So, the used data in the training of the model is a preprocessed data. Therefore, any incoming incident should be processed in the same way. Otherwise, the classification model will fail in classifying the incident and will not be able to map it to the correct class.

2. **Incidents Preprocessing**

   The preprocessing phase, which is a critical step in the contribution described in the clustering part, will remain the same when moving from historical time series data analysis toward online time series data classification. Hence, the data generated by the probes should be treated and prepared well, without changing or manipulating any fact in it, to be classified later on. Thus, this led us to perform the same preprocessing techniques described earlier in the clustering phase.

   Probes are generating data regularly and periodically. Each minute the probe will capture the reading of the Gamma dose rate in the environment. Readings are grouped in time order for each probe. When the reading exceeds the peak value for 30 minutes, the system in the REWS station will extract this as incident from the time it exceeded the maximum background mean till reaching the duration of 30 minutes above the peak. This extracted incident should be identified as fast as possible to check if it is an innocent incident caused by none harmful events like rain or storm, or it is a result of a real

threat, which led for the existence of high gamma dose radiation which is critical so that required actions should be taken accordingly. Therefore, to be able to classify this incident, it should be processed the same way we processed the incidents among the previously introduced clustering model. Otherwise, the system will not recognize this incident and the online classification process will not be beneficial.

Now we have an incident that should be classified, a preprocessing engine will be implemented for accepting raw data incident as an input and apply on it the two essential preprocessing techniques which are Z-normalization and Zero padding. Z-normalization is a must, because the classification model is trained by the dataset of phase one (clustering model phase) that is normalized. And as for the zero padding, a minimum incident size should be given for the classifier as an input. Thus, zero padding technique will be used to confirm the minimum required length. All those techniques will guarantee the best performance of the classification model when predicting the class of the current incident.

- **Data Normalization:** One of the challenging and most significant issues about our incident's data is that it is composed of values that are of different scale and amplitude. For phase one (clustering phase) to be completed successfully, it was a must to normalize our data so that we could have incidents all on the same level. The hard part that it was important to investigate in is what type or technique of normalization we should adopt for this process. By looking into the suggested solutions the best one that fitted our data is z-normalization [64]. Why? Because, other normalization techniques rely on factors or variables that do not exist in our data and if applied it will cause the loss of the real amplitude values of an incident. For example normalization that depends on Min-Max and decimal scaling, in both cases the minimum or maximum values are unknown in our data and unifying incidents on a selected min max will case incidents to lose its true amplitude value. Therefore, z-normalization is the best practice in our case and it has been proven to be the optimal solution when dealing with Time Series Data, because when incidents are normalized the shape is left the same to be compared by. Z-normalization results in having the mean equals to 0 and standard deviation between 1 and -1, and advantage here is that different incident are easily compared.
- **Data padding:** Once again it is all related to the trained machine learning model that is used for classification. Before performing any classification process the classifier should learn from a dataset in order to conclude some rules that it will base on in the classification part. The dataset that the model has learned from is of complete incidents

and not part of an incident. When performing online classification we cannot wait until the incident is done then attempt to do classification of the incident, because as explained before the gamma dose radiation is very harmful and in case of real incident we should predict the cause as soon as possible to limit the casualties. Therefore, the received parts of currently occurring incident should be padded to a certain length for the classification model to perform better in its mapping job.

3. **Incidents Classification**

   Back to our case and as explained earlier, probes that are implemented in different locations all generates data continuously for a fixed period of time that is 1 minute. Therefore, for the last 10 years probes are sensing the gamma dose rate found in the environment and sending its reading every single minute. Time Series Data is any numerical variable that can measure over time by fixed period of time. Time Series Data falls in two categories, Univariate Time Series Data and Multivariate Time Series Data.

4.3.3 . Architecture of the Proposed Model

We are not seeking for perfection, the intended result is to reduce the errors and not eliminate all errors. Our data is significant, and challenging, removing all error is impossible. The reason why all of the previous classifiers are implemented and tested was because each one was successful in identifying a particular class. Here comes our approach in building an ensemble classification models, were each classifier will perform its prediction and the output will not just be the predicted class, but also the probability of each class upon which it concluded to select the class of higher probability. Therefore, each classifier will give its predictions values and a councilor is introduced to take all the predictions of all the classifiers then deduces what the majority has classified this new incoming incident. The Councilor has to choose one of two possible choices. The first is to see the majority of the votes of the classifiers and if the aggregated probability is high (above 80%) then directly the decision is to assign the incident to this particular class with the highest probability. The second choice, if the aggregated probability is low (less than 79%) then this new incident should be considered as a new class of incidents. Here comes the role of the experts in the domain to examine this new incident and attempt to identify the nature and the reasons behind this incident. Furthermore, this new incident could be a new shape for an existing class that the model haven't trained on it yet. So in all cases this new incident is an added value for the model in the future when re-training the classifiers on identifying such cases. The proposed classification model is represented in Figure 4.5

1. **Online Incidents Extraction Phase**

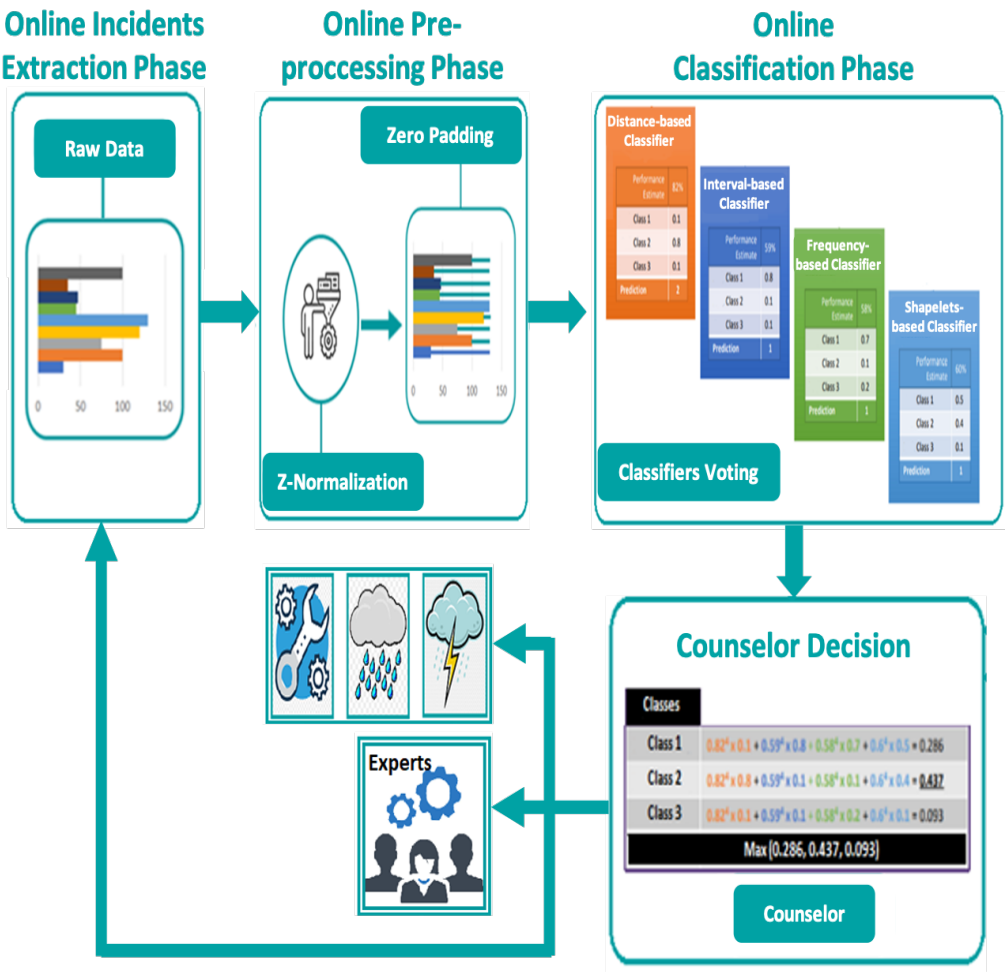

Figure 4.5: Proposed Classification Model

This is an online incident extraction phase. The data that is sent by the probes from different locations are continuously monitored. A high reading that is above the peak threshold will trigger the system to check if the reading of this probe will remain above the peak for 30 minute. If the incoming readings remain high (above the peak threshold) this series will be extracted as incident starting from the value above the maximum background mean till the 30 minute that has passed.

2. **Online Preprocessing Phase**

   Extracted incidents cannot enter directly to the classification phase. A pre-processing treatment for the raw incidents should be done. Incidents coming from different probes are of different characteristics in terms of length, scale and level. During the preprocessing phase the data of the incidents are normalized using z-normalization technique and padded using zero padding

technique. This preprocessing phase will not affect the shape of the incident; it will only standardize the incidents to become similar to the training dataset the classifiers have trained over. This will help classifiers to identify or predict the class of these unknown incidents.

3. **Online Classification Phase**

   The classification phase is divided into two phases. The first phase is the voting phase. In this phase four classifiers are implemented separately. Each classifier will accept as input the incoming incident. The four classifiers will run in parallel. The first classifier is Distance-based classifier implemented using K-nearest neighbor with Dynamic Time Warping + Barycenter averaging (DBA) as similarity measure. This classifier was able to successfully differentiate between calibration class and stormy class with accuracy reached 89.28%. The distance based classifier faced some difficulty in separating between rain class and stormy class, that why implementing Interval-based classifiers was needed. The Frequency-based classifier was built using Random Interval Spectral Ensemble RISE algorithm. This algorithm proven its ability to differentiate between rain class and stormy class, its accuracy reached 85.71%. As for the calibration class the algorithm had some slight errors were it classified calibration incidents as rain and vice versa. The third used algorithm in this approach is the Interval-based algorithm; it is similar to the Frequency-based algorithm except in the way it slices the series. Each series is split into intervals of varied length within the same decision tree, while RISE performs a random interval length splitting that varies from one decision tree and another, but within the same a fix interval is used. For the Interval-based algorithm, Time Series Forest classifier is implemented. This TSF classifier supports decision making especially between calibration class and rain class, its accuracy reached 89.28%. Finally, the last implemented algorithm is the Shaplets-based algorithm. Although the accuracy was low were it reached only 50%, but the significant of this algorithm is in differentiating between calibration class and rain class. The reason why this algorithm has failed in separating between the other classes is because of high similarity in the shape that some incidents of different class have. And since this algorithm is based on creating a bag of Shaplets (sub-shapes of the series) for each class to be used as discriminatory power, confusion happened.

   At the end, combining the classification ability of each classifier helped in overcoming the problems and challenges found in our data. Each classifier inside the classification phase will receive as input instance of the incoming incident. Then voting process will start, each classifier will output its prediction with probability of each class. Once all the classifiers are done, now comes the role of the councilor to aggregate the probability of each

class, then it will decide what the identity of this unknown incident is. Here counselor has three possible choices depending on the highest aggregated probability. If the probability remains high (above 80%) after three classification attempts, then the incident will be assigned to the class of the highest probability, else if the probability varies between 60% and 80 % then the incident will undergo further classification after collecting more incoming data readings. Moreover, the incident is left for the experts to check and verify if it didn't succeed in gaining a probability higher that 60%, so that it could be a new incident of a new class to be created, or new shape for existing class.

4.3.4 . Experimentation

In order to compare the different approaches of the state of the art, as well as to see the benefit of our proposed model, we decide to evaluate systematically different experiments and evaluations on the labeled time series data. After investigating all of the mentioned algorithms in the state of the art, we attempted to implement each classifier based on its best practice for selecting the optimal parameters and then apply it over our labeled data. First, the data we have is split into two sets, a training dataset (90% of the data set) and a testing dataset (forms the rest 10%). When splitting the data between training and testing, we guarantee that the training dataset is balanced and presented well in all three classes so that the classifier will be trained well.

All the implementations are done using python libraries. For the best environment performance and for easy implementation, we installed Anaconda, in which we used Jupyter notebook for writing and testing the code. Anaconda provides us with an isolated environment containing all the needed libraries to perform our tests. Going deeply into the libraries dedicated to Time Series Data, several methods are defined to handle this type of data. The traditional machine learning algorithms implemented for tabular classifiers cannot be applied in our case of time series data because it neglects the time factor essential in our data. So, besides Sklearn, pandas, numpy, and other libraries, we installed and used the Sktime library which contains the Time Series Classifiers.

In Table 4.1, we found the evaluation results for the four times series classifiers of the state of the art. Class 0 corresponds to the calibration cluster, Class 1 to the rain class and class 2 to the stormy rain class.

The first classifier is the KNN with DTW, which is a distance-based classifier. By default, this classifier uses the euclidean distance measure [18] to determine the membership of a class. For our case, the time series data requires a different metric algorithm because incidents are of varying length and are not perfectly aligned in time. Although the accuracy was not bad (89.28%), after training and testing it, some errors still occurred. By investigating what the model has failed, we deduced that it could detect the calibration class and the rain class but failed to identify the stormy rain class. The model got confused between rain class and stormy rain

class incidents and classified the stormy rain as rain incidents.

The second classifier is the Time Series Forest classifier which relies on the Interval based algorithm. This classifier depends on the information retrieved from various intervals of a series. At first, the classifier splits the series into random intervals; each has a random starting point and length. Then the algorithm extracts summary features (Slope, Mean, and Standard Deviation) from those intervals. The extracted features form the feature vector representing the interval. Since this algorithm is based on the Random Forest algorithm, it will construct and train a decision tree from the extracted features. Several trees are constructed to support decision-making and select the majority of the trees in the forest. After training and testing the TSF classification model, the performance was good (89.3%), but not enough. The model was able to identify the calibration class but it faced some errors when identifying the stormy rain class.

Table 4.1: Applying different classification algorithms on the dataset

Classification Algorithm	Accuracy	Distinguished Classes	
		Pass	Fail
KNN+DTW	89.28%	Class 0 & Class 1 Class 0 & Class 2	Class 1 & Class 2
TSF	89.3%	Class 0 & Class 2	Class 0 & Class 1 Class 1 & Class 2
RISE	85.71%	Class 1 & Class 2	Class 0 & Class 1 Class 0 & Class 2
Shapelet	50%	Class 0 & Class 2	Class 0 & Class 1 Class 1 & Class 2

The third classifier is the Random Interval Spectral Ensemble (RISE) classifier. This classifier is based on the frequency features extracted from the series after splitting it into intervals. It sounds similar to the previous classifier, the TSF, especially because it also uses the Random Forest algorithm. It differs from TSF in two ways. First is how it splits the series into intervals, where the intervals for each decision tree are of the same length. The second difference is in the type of features the algorithm extracts from the intervals, where RISE extracts spectral features (series-to-series features) and not summary statistics. The algorithm was significant in classifying the rain class from the data. While in the rest of the classes, it faced some errors. The Accuracy of this model reached (85.71%).

Finally, the last classifier is the shapelets based classifier. This classifier is very popular and used when dealing with time series data. A Shapelet is a sub-shape of a series. A bag of shapelets is used to represent a particular class. When extracting those Shapelets, the algorithm searches for shapes with discriminatory power to identify a class. Shapelets form the identity of each class. When a new unknown incident arrives, the algorithm will extract its shapelets and compare

them to the classes' shapelets to confirm which class the incident belongs to. The Shaplete based algorithm was implemented and tested over our data, but results came unsatisfying. After several tests and attempts to enhance the model's overall accuracy, it nearly reached 50%. However, after we investigated the results, we uncovered the reason for such an outcome. The data that we have is very challenging because it is very similar to each other, which makes its shapes very similar; this is why the model got confused. Even though the classification model was able to identify the Rain class but failed in the other two classes (Calibration and Stormy Rain).

To start evaluating our online classification module, we first tested incoming incidents. These incidents are online preprocessed after being extracted and then prepared to be classified with the classification algorithms. Next, the classification algorithms will work individually in parallel on the incoming incident trying to classify it as soon as possible. The classification algorithms presented in Table 4.2 will return their predictions for the incoming incidents as soon as possible. This prediction will be in the form of probability suggested by each algorithm to each incident while trying to map it to the respective class.

Table 4.2: Testing our Councilor approach

	KNN+DTW	TSF	RISE	Shapelet	Counselor Performance	Counselor Decision
Incident 1	C0 (100%) C1 (0%) C2 (0%)	C0 (95%) C1 (0%) C2 (5%)	C0 (63%) C1 (6%) C2 (31%)	C0 (34%) C1 (21%) C2 (45%)	C0 (73%) C1 (6.75%) C2 (20.25%)	Wait for more data
Incident 2	C0 (98%) C1 (2%) C2 (0%)	C0 (96%) C1 (1%) C2 (3%)	C0 (68%) C1 (23%) C2 (9%)	C0 (99%) C1 (0%) C2 (1%)	C0 (90.25%) C1 (6.5%) C2 (3.25%)	C0
Incident 3	C0 (0%) C1 (31%) C2 (69%)	C0 (0%) C1 (0%) C2 (100%)	C0 (0%) C1 (3%) C2 (97%)	C0 (0%) C1 (0%) C2 (100%)	C0 (0%) C1 (8.5%) C2 (91.5%)	C2

Finally, the counselor will start performing the task assigned to it. Thus, the role of the counselor will be to decide which algorithm acts the best and gives the perfect prediction for the incoming incident as shown in Table 4.2 where Incidents 2 and 3 were assigned to classes 0 and 2 respectively. However, incident 1's probability was not enough for the counselor to make a decision that it is why it suggested to wait for more data.

4.3.5 . Evaluation

The proposed model overcomes the issue that we were concerned about. The problem with each classification algorithm when tested on its own was its ability to identify a single class and failing in differentiating between the rests of the classes. By combining the outputs of those four algorithms, the councilor was able to either commit what is the identity of the unknown incoming incident or consider it as a

new incident that is related to in new class to be examined by experts. Therefore, the proposed classification model output was satisfying and it supported decision making for predicting the class of the incoming incidents.

4.4 . Conclusion

In this work, we presented our machine learning based framework for autonomously identifying the causes behind the online incoming incidents caused by high gamma dose rate readings. After extracting, preprocessing, and clustering the historical incidents, our approach is to apply a machine learning model that will match online incoming incidents to their similar clustered ones to identify the causes behind them as soon as possible using supervised classification.

In the classification phase, we, specifically, tackled the problem of classifying time series using several classification algorithms at the same time which was properly addressed nowhere in the literature. We researched and experimented with the different classic and state of the art approaches to evaluate their compatibility. When those approaches failed to classify our data when testing each approach alone properly, we proposed our Counselor Classification Model for using all the classification algorithms simultaneously and voting for the one with the best outcome.

Displaying the obtained matching percentages through the experimentation of various algorithms, we were able to highlight how our model gave the best results comparatively. Also, the experts expressed positive and hopeful thoughts inspecting the results which motivated us to publish this contribution in an article. As future work, the next step would be to improve the quality of the overall framework by exploring the evaluation with more data sets to automate the evaluation as well.

5 - Prototype

Contents

5.1 . Concept and Principals

In Chapter 2 of this dissertation, we highlighted the need of moving towards an Intelligent Radiation Early Warning System. To achieve our goal, we presented the proposed RIMI framework which will handle this task through two main phases.

As discussed in Chapters 3 and 4, combining the obtained data with machine learning algorithms form the backbone of our intelligent system. Hence, a design methodology that can handle the different challenges behind the traditional system is highly favorable. The system design should take into consideration (i) dealing with the data generated from historical databases or collected online from the probes, (ii) preprocessing the collected data and prepare it for the analysis process, and (iii) analyzing the preprocessed data in order to extract and test the intelligence behind the proposed system.

In this chapter we will present the prototype used to develop our methodology. We will introduce the software components and implementation of the system architecture in order to put it in practice. Thus, the objective is to develop a software or a running system to manage the introduced contributions behind this dissertation using different components. Moreover, we will introduce the technicalities, parameters, and some algorithms used to reach our objectives.

5.2 . System Architecture

Since the RIMI framework is developed through two phases, we will focus in this section on the main components that are behind this framework. As mentioned earlier, this framework aims to replace a human-driven verification system to refine the peak values and alerts and detect the causes behind them automatically with high accuracy guaranteed.

The proposed architecture in Figure 5.1 highlights the discussed components in Chapters 3 and 4 in order to introduce a reliable and scalable ecosystem that can automate the data analysis pipelines. These components communicate with each other in a synchronous matter using a messaging tube between them. The system architecture provides a road map from data collection (such as historical data and real time data), to data preprocessing (such as incident extraction, data imputation, normalization, and padding), data clustering (using DTW similarity measure, K-means with DBA clustering algorithm, and obtaining the optimal number of clusters), data classification (using several classification algorithms, voting for the classifiers, and the counselor decision), and the results obtained through clusters labeling (offline environment) or incident identification (online environment).

5.2.1 . Common Components

As we can notice from the system architecture in Figure 5.1, the collected data should pass through several components before selecting its path through clustering or classification environments. We referred to those components as

common components as they are initial for the data before proceeding with the desired environment.

The common components introduced in Figure 5.1 are the Data Collector and the Data Preprocessing components.

1. Data Collector

   There are different types of data used in the Radiation Early Warning System. The data typically originates from the probes are the gamma dose rate data and the internal factors data. We defined data collection strategies based on the nature of the data and their uses. The historical data will be collected in batches mainly from the running REWS databases along with the internal factors data while during the real time collection, the data are collected online and sent to other components to be analyzed automatically by the system. Therefore, the Data Collector component has to deal with two aspects. The batch collection which deals with data generated from the historical databases and the real time data which is generated from the probes.

   Knowing that the gamma dose rate time series data was one of the important factors to perform this research, we investigated two data sources before reaching the objectives behind this dissertation. Thus, we started with the Radiation Early Warning System used by the Lebanese Atomic Energy Commission. After performing several preprocessing methods and preparing the data for analysis, we noticed the data was not clear enough for us to apply our tests on. Since the Radiation Early Warning System in Lebanon consists of 29 probes only, so the collected data was not enough to proceed with proposed system architecture.

   Next, we moved searching for another data sources until the Federal Office for Radiation Protection (BfS) provided us with their Radiation Early Warning System data. The data was composed of gamma dose rate time series which represents the minute-by-minute gamma dose rate readings. The Germany Radiation Early Warning System consists of more than a thousand probes where the experts provided us with the past 10 years data (around 5 GB) monitored by these probes. The data monitored by each probe was saved individually in single files where each file represents the data monitored by a specific probe during a specific year.

   Moreover, the probes are grouped into 9 divisions where each division is dedicated to monitor a specific area in Germany. Knowing that the groups of probes forming divisions are placed near to each other, so it is normally that they share the same behavior upon an increase in the gamma dose rate level at a specific area. Thus, through this research, we chose 45 probes randomly from different divisions to make sure that we covered all the areas monitored by the Radiation Early Warning System in Germany. Moreover,

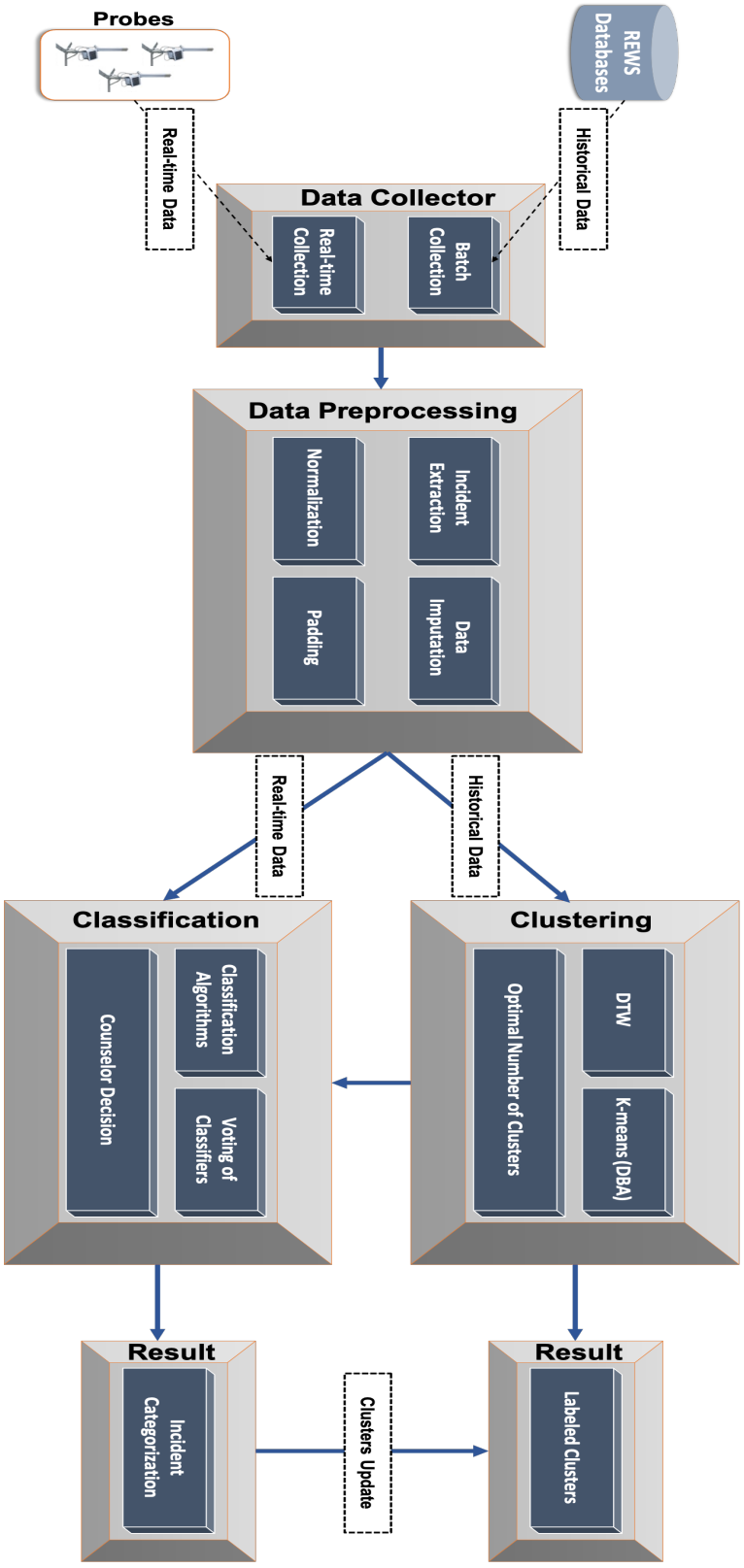

Figure 5.1: System architecture

this will improve the accuracy of the proposed model since the gamma dose rate analyzed is chosen from probes placed in different areas and exposed to different conditions.

2. Data Preprocessing

   Before any analysis phase, a lot of time have to be spent studying the data to find the proper preprocessing algorithms to apply [65] [66]. The preprocessing algorithms are to ensure the quality and cleanliness of the data before inserting it into any analysis algorithm. Inspecting the data, the major observations were: the incident extraction process, the vastly varying length issue of the incident, and the different scale issue of the incidents. Therefore, the second common component which is the Data Preprocessing component explains in details how the incidents are captured and extracted. Furthermore, it clarifies the preprocessing methods used to deal with different types of incidents.

   (a) Incident Extraction

   Incident Extraction service depends on several parameters to perform the detection and the extraction of the incidents. In this step we shall explain how different parameters are calculated so that they are used to deal with incidents from the historical database and with online incoming incidents. The parameters are: background mean, background interval, and the peak value.
   It is important to note that the mean and the peak values are given values at the beginning of the system, but, again, these value are not fixed; they shall be refined over time to better suit the default (normal) gamma rate of the location at which the probe is installed.

   In what follows consider the following definitions:

   - Let S be the set of all background values
   - Let $P \subset S$ be the set of all background values $\geqslant$ peak value. Let's call it set of peak values.
   - Let $S_{i,j}$ be the subsequence of S from index i to j, n be the number of background values in the specified subsequence, and pk the number of peak values.

   In order to calculate our parameters, we applied the concept of sliding window by calculating them per month as we observed that the difference between the values from month to month are not changing drastically. This computation model produces a catalog with the corresponding parameters for each month to be used in the later steps.

i. **Background:** The background mean is the default gamma rate. This value evolves over time so it must be evaluated regularly. It is calculated as follows:

$$\bar{b}_{i,j} = \frac{\sum_{m=1}^{n-pk} b_m}{n - pk}$$

$$\mid b_m \in S_{i,j} \; and \; b_m \notin P_{i,j}$$

The peak values are excluded as they are considered abnormal values.

ii. **Background Interval:** The background interval B is the range of the *safe* gamma dose rate values in the environment. We researched several methods to find the most suitable for determining the *lower* and *upper* bounds of the interval annotated $lowerB$ and $upperB$ respectively. We found that the *mean* and *standard deviation* are the most promising to find the background level interval. It has been chosen due to the nature of our data; uniformly distributed.
The standard deviation is calculated at the end of each month using the background mean calculated earlier. It is calculated as follows:

$$\sigma_{i,j}^2 = \frac{\sum_{m=1}^{n-pk} (b_m - \bar{b}_{i,j})^2}{n - pk}$$

The background interval is thus calculated as follows:

$$lowerB_{i,j} = \bar{b}_{i,j} - \sigma_{i,j}$$

$$upperB_{i,j} = \bar{b}_{i,j} + \sigma_{i,j}$$

$$B_{i,j} = [lowerB_{i,j}, upperB_{i,j}]$$

iii. **Peak Value:** This value is the value that distinguishes incidents. Once a value exceeds it, an alarm is triggered. Experts in different countries depend on different methods. Some considered it 1.5 times the mean, others considered it 2 times the mean. We chose to go with the former method as it is more precise knowing that this value can be changed to suit experts' expectations. It is calculated simply as follows:

$$peak_{i,j} = \bar{b}_{i,j} \times 1.5$$

In the following, we assume that the computation model catalog is fully computed on historical and real time data before the extraction starts.

An incident is extracted I if it is a sub-sequence of the original time series S which starts from a value within the background interval and reaches the peak at least one time before going back to the background interval.

$$I_{s,t} \subset S_{i,j}$$

$$I_{s,t} = \{b_k \mid s \leq k \leq t \ , \ \{b_s, b_t\} \in B_{i,j},$$

$$\forall k \in [s+1, k-1] \ k \notin B_{i,j}, \ and \ \exists k \geqslant peak_{i,j}\}$$

Extracting the incidents from the historical databases was not an easy task for us. We relied on the parameters used by the experts and modified them in order to create a catalog of parameters that can fit the extraction process perfectly. The extraction step will be done during both phases of the RIMI framework (offline extraction and online extraction).

During offline extraction, the running system will rely on the formerly mentioned catalog of computed values as well as the $I_{s,t}$ definition to properly extract all the incidents.

The needed parameters for the incident extraction process will be calculated based on the historical data provided. Starting with the first month's data, this data will be used for bootstrapping where incident extraction will not take place for this month. So, at the end of this month, the background average will be calculated which will result in calculating the peak value for this month. Then, we will go again through the data for the first month to find the background interval which represents the range of safe values of gamma radiation. We explored several methods to find the most suitable one that determines the lower and the upper bounds of the background interval. Our study revealed that the standard deviation [67] is promising to find the background level interval. We choose standard deviation because of the nature of the distributions of data. According to our observation, radiation level data are uniformly distributed and to the best of our understanding standard deviation is a suitable technique for finding intervals when data are uniformly distributed in a two-dimensional graph. The standard deviation will be used for finding the background interval after excluding the peak values from this calculation. This will result in obtaining the background interval, the peak value, and the background average from the data of the first month in the historical data. For the following months, these parameters will be calculated at the beginning of each month based on the previous month's data. This will form a catalog indicating the parameters that will be used each month for the extraction process.

Next, the extraction process will start searching for peak values to take place. The algorithm will run on a full time series and will compare the values based on their related dates with the corresponding parameters in the catalog. Once a peak value is spotted, its corresponding date will be the indication for which parameters to be used from the parameters catalog. The incident will be extracted once a peak value is captured starting from the nearest value before the peak value which belongs to the background interval and ending at the nearest value after the peak value which belongs to the same interval.

Later, after extracting the incidents from the historical databases, they will undergo several preprocessing methods and algorithms in order to start forming classes of incidents based on the situations discovered. Algorithm 1 describes the process of extraction.

As can be seen in Algorithm 1, initially, a background mean m and peak value p have to be supplied along with the time series data S of a probe.
The time series data S is grouped by month to form the sub-sequences *months* in order to implement the sliding window concept. At each *month*, the past mean $\bar{b}$ and *peak* values are used to calculate the standard deviation σ, the background interval $(lowerB, upperB)$, as well as the set P of peak values.
For each peak value, the starting point s and end t are obtained. They are the point at which the sub-sequence leaves and re-enters the background interval respectively.
Our approach introduces a locking mechanism which ensures that whenever a peak value is detected any peak value that is encountered before re-entering the background interval is excluded so as to avoid redundancy.

The next stage will be the online extraction of the evolving incidents. At the beginning of each month, all the parameters will be presented based on the previous month's data. During the online extraction, two processes will be occurring at the same time. The first process is the incremental calculation for each of the parameters that will be used for the next month's analysis so that they will be presented at the end of the current month. The second process will be the analysis process that will take place based on the presented parameters from the previous month's data. Therefore, once the gamma dose rate reached the peak value and remains for more than 30 minutes above it, the online incident extraction will start from the nearest previous reading that belongs to the background interval and extract the current incident occurring referring to the parameters catalog that is updated each month.

Algorithm 1 Extract incident fragments

Require: a time series: S; initial mean: m; initial peak value: p;
Ensure: set of incident fragments

$incident\_fragments \leftarrow empty$
$\bar{b} \leftarrow m$
$peak \leftarrow p$
$months \leftarrow$ divide S into subsequences by months
for $month$ in $months$ **do**
  $\sigma \leftarrow$ standard deviation of values where $b \leq peak$
  $lowerB \leftarrow \bar{b} - \sigma$
  $upperB \leftarrow \bar{b} + \sigma$
  $P \leftarrow b$ in S where $b \geqslant peak$
  $excluded\_peaks \leftarrow empty$
  for $peak\_value$ in P **do**
    if $peak\_value$ not in $excluded\_peaks$ **then**
      $s \leftarrow peak\_value$ (starting point of incident)
      repeat
        $s = s - 1$
      until $(s \leq upperB$ and $s \geqslant lowerB)$ or s = start of $month$
      $t \leftarrow peak\_value$ (ending point of incident)
      repeat
        $t = s + 1$
        if $t \geqslant peak$ **then**
          $excluded\_peaks$.add(t)
        end if
      until $(t \leq upperB$ and $t \geqslant lowerB)$ or t = end of $month$
      $incident\_fragments$.add($month$[s:t])
    end if
  end for
  $\bar{b} \leftarrow$ mean of values where $b \leq peak$
  $peak \leftarrow \bar{b} \times 1.5$
end for
return $incident\_fragments$

As mentioned before, during our research, we considered only the incidents that remains for more than 30 minutes since incidents shorter than 30 minutes do not form critical effects on the environment. Moreover, incidents that are less than 30 minutes are usually generated because of internal factors error or other innocent causes mentioned previously in Chapter 2.

(b) Data Imputation

The main task behind this service has to deal with the missing data problem. As the sent gamma dose rate readings may be interrupted by any factor resulting in not sending the data every minute, this service will handle the missing data issue by dropping the whole time series data for batch collection or dropping the extracted incident if it has some missing data points and start the extraction process again.

(c) Normalization

Since the probes depend on the background and peak values to capture incidents, and since these values can evolve based on the location and the age of the probe, it was normal for us to have incidents with different scales and levels.

In the beginning, we were content with subtracting the background mean from the incidents in order to just have them on the same level because we believed that any normalization or standardization method will have us losing the real amplitude values reached which we believed may be critical in our identification process. Following this approach, we were not able to give meaningful results because of the sometimes huge distance between the incidents.

Through this service, we rely on the Z-normalization method which is so promising in normalizing the values while the shape of the incident will remain the same. Applying this service, we make sure that the extracted incidents will be with the same scale and level. This service will apply a zero padding on the normalized incidents to make sure that the mean and the standard deviation of the data will not be affected.

After applying this service, we will make sure that all the extracted incidents are of the same scale and length.

(d) Padding

As explained in Chapter 2, different factors that affect the gamma dose rate readings act in different matter. This was clear for us that the extracted incidents will not be with the same length since the duration of the incidents will differ with respect to the running cause.

To deal with this issue, we call the Padding service through the Data Preprocessing component. After fixing the scale and level issue by

applying the Z-normalization service on the incidents, the padding service needs to be applied to deal with the length issue.

5.2.2 . Offline Environment

The offline environment represents the data analysis process for the batch data collected offline from the historical database. This environment plays a vital role in building the predictive model behind the RIMI framework. Figure 5.2 represents the implementation prototype of the offline environment where several components and services are combined to perform the learning methodology for the RIMI framework.

Through the learning methodology, the data should be collected from the historical databases in order to study the behavior and the attitude of the stored data. Once the data is collected, the incident extraction service will start based on catalog of parameters prepared earlier. This step will result in extracting all the incidents from the historical data.

Next, the system should deal with the varied extracted incidents since they are of different scale and length. Starting with the different scale problem, the Z-normalization service should start tackling this issue to make sure that all the incidents become at the same level. Then, the zero padding service will start addressing the different length problem. This will result in having all the incidents with the same length.

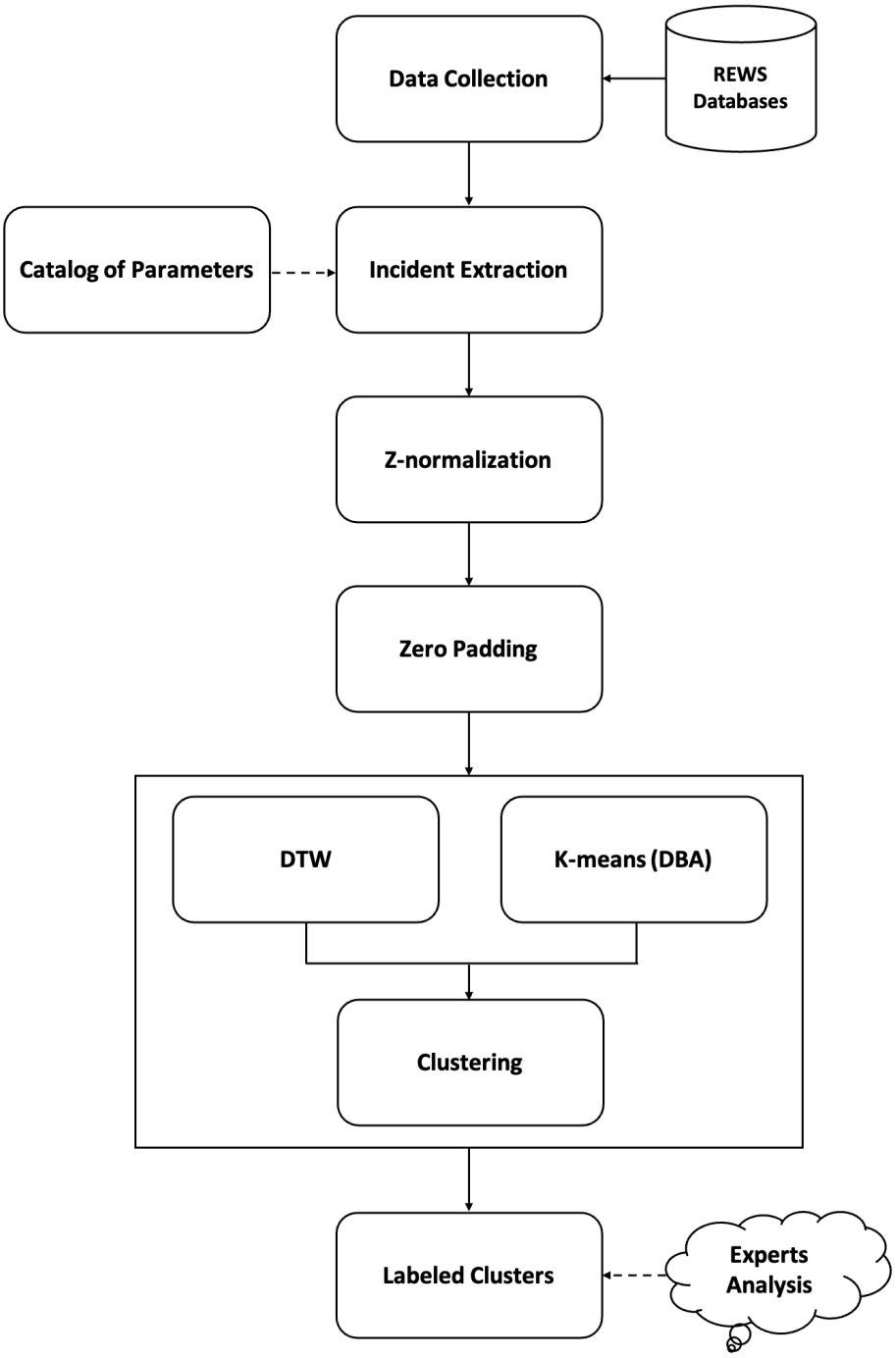

Figure 5.2: Offline environment implementation

Since the extracted incidents become with the same scale and length, then the clustering process will be triggered. The clustering component will take place by combining the DTW similarity measure along with the K-means clustering algorithm equipped with DTW Barycenter Averaging method. The clustering process

will result in different clusters where the clusters' labeling will be performed with the help of the experts in order to make sure that they can identify the real situation behind each cluster.

5.2.3 . Online Environment

The online environment represents the real time detection and prediction processes for the real time data generated from the probes. This environment formulate the second phase behind the RIMI framework. Several components are combined to perform the detection and prediction tasks behind the online environment as shown in Figure 5.3 where the implementation prototype of the online environment is represented.

Through the online environment, the data are collected from the probes at real time. Referring to prepared catalog of parameters, the incident extraction service will start once the gamma dose rate readings reach the peak level and remain for more than 30 minutes above the peak. After 30 minutes, the incident extraction service will start extracting the current incident subsequence.

Next, the Z-normalization service followed by the zero padding service will start fixing the scale and the length issues of the current extracted incident. This will put the current extracted incident at the same scale and length with those clustered in the offline environment.

After that, the classification algorithms will start trying to classify the incoming incident to one of the obtained clusters from the offline environment. Each classification classifier will vote for the accuracy of detecting the related cluster. Then, the counselor decision service will run to choose the best fit classification algorithm along with its result referring to its accuracy percentage. If the percentage was high enough for the counselor, then the counselor will categorize the incoming incident to the related suggested cluster by the classification algorithm. Otherwise, the counselor will ask the data collector for more gamma dose rate readings in order to rerun the environment services again which may clarify the situation more.

However, if all the classifiers assign a low accuracy percentages, then this may indicate that a new scenario is happening which needs experts verification in order to form a new cluster representing this scenario. This continuous learning process will update the clusters formed in the offline environment.

5.3 . Technicalities

To realize the methods and algorithms required to develop the different components of the RIMI framework, a machine with a configuration of Intel Core i7 processor and 32 GB of RAM running windows 10 was used. While developing the approach behind the first phase of RIMI framework which is Building the Predictive Model phase, the experiments were done using Python language in the Visual Studio Code environment. In order to be able to implement the algorithms we used

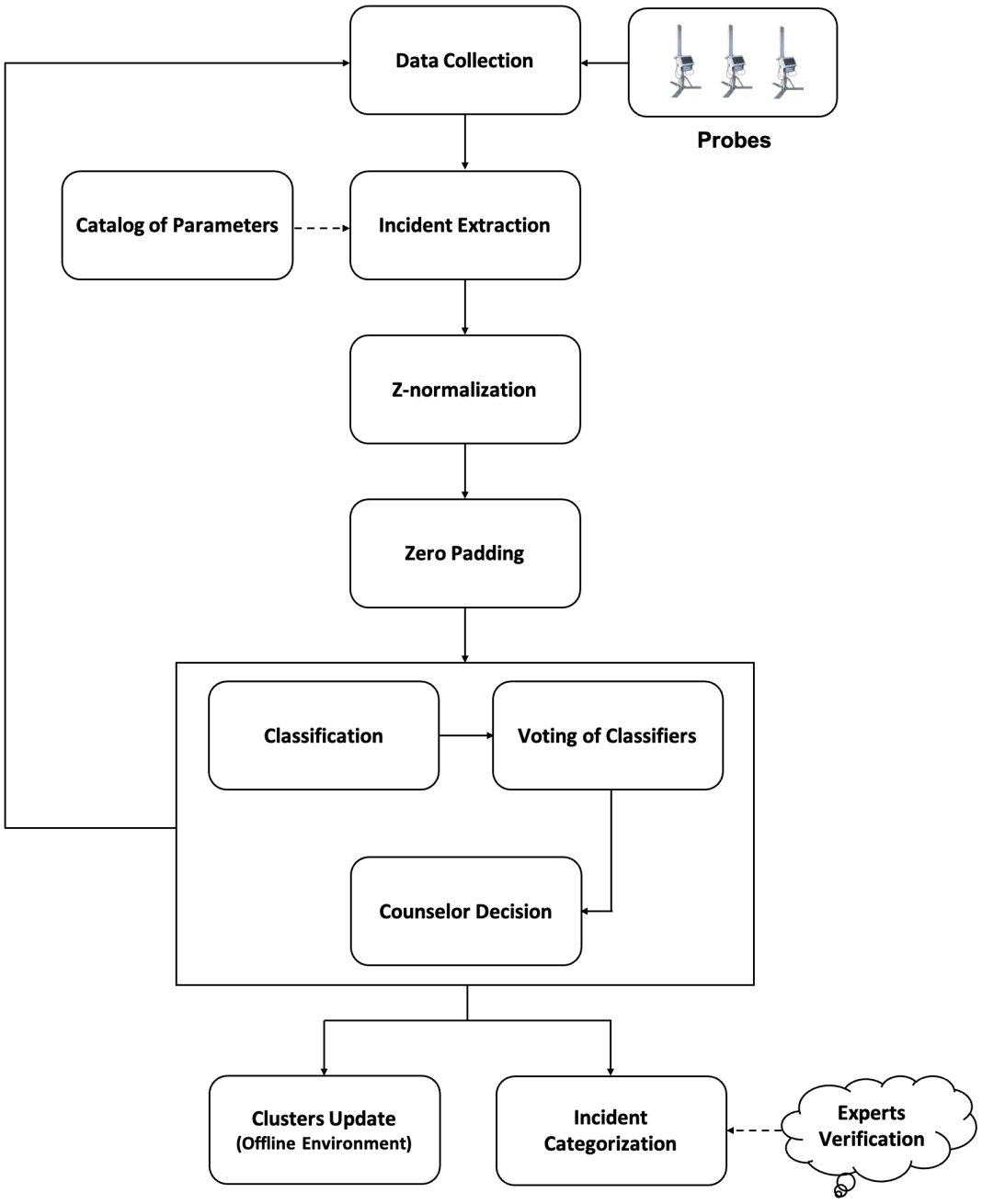

Figure 5.3: Online environment implementation

the implementations provided by the popular machine learning libraries (sklearn, tslearn, dtaidistance..). However, we couldn't implement some of the models using the previous libraries especially when the samples were of unequal length, which is a requirement in the methods provided, so we had to implement the methods by ourselves using as much as we could from these libraries.

Through developing the approach behind the second phase of RIMI framework which is the Online Prediction, all the implementations are done using python libraries. For the best environment performance and for easy implementation, we installed Anaconda, in which we used Jupyter notebook for writing and testing the code. Anaconda provides us with isolated environment containing all the needed libraries to perform our tests. Going deeply into the libraries dedicated for Time Series Data, there exist a number of methods that are defined to handle this type of data. The traditional machine learning algorithms that are implemented for tabular classifiers cannot be applied in our case of Time Series Data because it neglects the Time factor which is essential in our data. So, beside Sklearn, pandas, numpy, and other libraries we installed and used Sktime library which contains the Time Series Classifiers.

6 - Conclusion

Contents

Through this chapter, we will summarize the work presented throughout the previous chapters, and emphasize our achievements against the research questions presented in Chapter 3 and Chapter 4 in Section 6.1. Thereby, we will highlight possible future research directions in Section 6.2.

6.1 . Summary of Contributions

We started this dissertation by introducing the concept of monitoring the radiation level in the environment. Then, we introduced the Radiation Early Warning System which is the system used by many countries along the world to perform this task. However, we listed the shortcomings behind the REWS and highlighted the idea of changing the current system into an intelligent one. Thus, in Chapter 2, we presented our main approach through an end-to-end framework titled RIMI for preprocessing, processing and analysis of radiation level data. The objective of developing this framework is to eliminate the manual intervention in radiation level monitoring systems.

Through this dissertation, we explained the two main phases of the presented framework: Building the Predictive Model and Online Detection and Prediction. The key components of the RIMI framework included incident extraction and preprocessing, incident clustering, and online incident classification. We provided a detailed description of each phase of the framework aiming to introduce an intelligent system that can helps the experts in faster decision making regarding verification of an alarm during an incident. When this model was proposed to the experts, we got their approval after exploring its results. Thus, the tool is ready to be deployed next to the traditional REWS to get more feedback. Moreover, we have broken down the RIMI framework components into two research questions that this thesis work aims to answer.

Research Question 1

What is the best-fit unsupervised machine learning model that should be used for clustering time series data of varying lengths and different scales?

To answer this question, we presented in Chapter 3 our machine learning based framework for autonomously identifying underlying events behind high gamma dose rate historical incidents. Our approach was to, after extracting and preprocessing the extracted incidents, apply a machine learning model that will group similarly behaving incidents as caused by the same underlying event using unsupervised clustering. The groups are then evaluated by the experts to recognize the events and, hence, label the incidents.

The proposed framework was divided into four different phases: Incident Extraction, Data Preprocessing, Time Series Clustering, and Cluster Evaluation. The first two phases we did as a first step to prepare the data set for the machine

learning model. In the next step we performed the two phases of clustering and evaluating.

In the clustering phase, we, specifically, tackled the problem of clustering time series of varying length which was properly addressed no where in the literature. We researched and experimented with the different classic and state of art approaches to evaluate their compatibility. When those approaches failed to properly cluster our data we proposed our specific model for clustering the sequences of varying length.

Displaying the obtained clusters through the experimentation of various models, we were able to highlight how our model gave the best results comparatively. Also, the experts expressed positive and hopeful thoughts inspecting the results which motivates us to continue the work towards the intelligent REWS.

Research Question 2

What is the machine learning model that should be used for online time series classification of special behavior and how do the different models perform in practice?

To answer this question, we presented our machine learning based framework for autonomously identifying the causes behind the online incoming incidents caused by high gamma dose rate readings. Our approach was to, after extracting, preprocessing, and clustering the historical incidents, apply a machine learning model that will match online incoming incidents to their similar clustered ones to identify the causes behind them as soon as possible using supervised classification.

The proposed framework was divided into four different phases: Incident Identification, Data Preprocessing, Time Series Classification, and Classification Model Evaluation. The first two phases we did as a first step to prepare the incoming data set for the machine learning model. In the next step we performed the two phases of classifying and evaluating.

In the classification phase, we, specifically, tackled the problem of classifying time series using several classification algorithms at the same time which was properly addressed no where in the literature. We researched and experimented with the different classic and state of art approaches to evaluate their compatibility. When those approaches failed to properly classify our data when testing each approach alone, we proposed our Counselor Classification Model for using all the classification algorithms at the same time and voting for the one with the best outcome.

Displaying the obtained matching percentages through the experimentation of various algorithms, we were able to highlight how our model gave the best results comparatively.

Furthermore, we were able to tackle the problem statements and the research problem behind this dissertation using the only available data in the historical databases. The unlabeled gamma dose rate time series data was sufficient for us

to reach our objective behind this dissertation. Moreover, the experts expressed positive and hopeful thoughts inspecting the results.

6.2 . Future Work

As explained in Chapter 2, the main objective behind our approach is to change the current Radiation Early Warning System into an intelligent automated system. After introducing our contributions, we can state some objectives that can be handled later as a future work.

First, the experiments performed during the first phase were not considering all the probes due to the limitations of time and resources. Thus, the extracted incidents represented the scenarios captured by several random probes. However, this process can be extended to cover all the probes so that the obtained extracted incidents can help in producing more accurate clusters with more causes affecting the gamma dose rate.

Moreover, another objective that can be handled as a future work is to change the current system into a full automated system. Although the presented contributions will help in capturing the incoming incidents and identify them as soon as possible, however, the system still needs the experts verification at the end of the incident identification process since the running situation is critical and to make sure that the intelligent system is performing well. Thus, the next step would be to work on improving the quality of the overall framework by exploring the evaluation with more data sets in order to automate the evaluation as well. This will help the system to be highly qualified to capture the real cause behind the current incoming incident with high accuracy without referring to the experts.

Bibliography

[1] Agnieszka Jelewska and Michał Krawczak. "The Spectrality of Nuclear Catastrophe: The Case of Chernobyl". In: *Electronic Visualisation and the Arts* (2018). doi: 10.14236/ewic/evac18.30.

[2] International Atomic Energy Agency. *The Fukushima Daiichi Accident*. International Atomic Energy Agency, 2015. isbn: 9789201070159. url: https://inis.iaea.org/search/search.aspx?orig_q=RN:46110858.

[3] P. Senin. *Z-normalization of time series*. url: https://jmotif.github.io/sax-vsm_site/morea/algorithm/znorm.html (visited on 03/07/2022).

[4] W. B. Mann. "The international chernobyl project technical report: Assessment of radiological consequences and evaluation of protective measures". In: *Applied Radiation and Isotopes* 44.6 (1993), pp. 985–988.

[5] Dang Quang Thieu et al. "Study, Design and Construction of an Early Warning Environmental Radiation Monitoring Station". In: *Communications in Physics* 22.4 (2012). issn: 0868-3166. doi: 10.15625/0868-3166/22/4/2665.

[6] Necati Küçükarslan et al. "Early Warning Environmental Radiation Monitoring System". In: Jan. 2005, pp. 33–41. isbn: 1-4020-2376-6. doi: 10.1007/1-4020-2378-2_7.

[7] Steven Biegalski et al. "Caribbean radiation early warning system (CREWS)". In: *Journal of Radioanalytical and Nuclear Chemistry* 248 (June 2001), pp. 637–642. doi: 10.1023/A%3A1010624224587.

[8] H. Dombrowski et al. "Recommendations to harmonize European early warning dosimetry network systems". In: *Journal of Instrumentation* 12 (Dec. 2017), P12024–P12024. doi: 10.1088/1748-0221/12/12/P12024.

[9] Ramon Casanovas et al. "Performance of data acceptance criteria over 50 months from an automatic real-time environmental radiation surveillance network". In: *Journal of environmental radioactivity* 102 (Aug. 2011), pp. 742–8. doi: 10.1016/j.jenvrad.2011.04.001.

[10] Satoshi Shimizutani and Hiroyuki Yamada. “Long-term consequences of the atomic bombing in Hiroshima”. In: *Journal of the Japanese and International Economies* 59 (Mar. 2021), p. 101119. doi: 10.1016/j.jjie.2020.101119.

[11] M. Farid et al. “Design of early warning system for nuclear preparedness case study at Serpong”. In: vol. 1862. July 2017, p. 030067. doi: 10.1063/1.4991171.

[12] *International Atomic Energy Agency*. url: https://www.iaea.org (visited on 03/07/2022).

[13] Patrick Kessler et al. “Novel spectrometers for environmental dose rate monitoring”. In: *Journal of Environmental Radioactivity* 187 (Feb. 2018). doi: 10.1016/j.jenvrad.2018.01.020.

[14] A. McGovern et al. “Identifying predictive multi-dimensional time series motifs: an application to severe weather prediction”. In: *Data Mining and Knowledge Discovery* 22.1-2 (2011), pp. 232–258.

[15] K. Eamonn et al. “Indexing large human-motion databases”. In: *13th Int’l Conf. on Very Large Data Bases, pages*. 2004, pp. 780–791.

[16] A. Mueen. “Time series motif discovery: dimensions and applications”. In: *WIREs Data Mining Knowl Discov* 4.2 (2014), pp. 152–159.

[17] B. Liu et al. “Efficient motif discovery for large-scale time series in healthcare”. In: *IEEE Transactions on Industrial Informatics* 11.3 (2015), pp. 583–590.

[18] J.C. Gower. “Properties of Euclidean and non-Euclidean distance matrices”. In: *Linear Algebra and its Applications* 67 (1985), pp. 81–97.

[19] C. S. Myers and L. R. Rabiner. “Connected digit recognition using a level-building DTW algorithm”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 29 (1981), pp. 351–363.

[20] L. Mico and J. Oncina. “Comparison of fast nearest neighbour classifiers for handwritten character recognition”. In: *Pattern Recognition Letters* 19.3 (1998), pp. 351–356.

[21] M. Vlachos, D. Gunopoulos, and G. Kollios. “Discovering similar multidimensional trajectories”. In: *18th International Conference on Data Engineering. IEEE Computer Society, Washington.* 2002.

[22] T. Liao. “Clustering of time series data - a survey”. In: *Pattern Recognition* 38.11 (2005), pp. 1857–1874.

[23] S. Rani, G. Sikka, and T. W. Liao. “Recent techniques of clustering of time series data: A survey”. In: *International Journal of Computer Applications* 52 (2012).

[24] A. Sarda-Espinosa. “Comparing time-series clustering algorithms in r using the dtwclust package”. In: *Manual of the R package dtwclust* (2017).

[25] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah. “Time-series clustering – a decade review”. In: *Information Systems* 53 (2015), pp. 16–38.

[26] L. Kaufman and P. J. Rousseeuw. “Finding Groups in Data: An Introduction to Cluster Analysis (1 ed.)” In: *New York: John Wiley* (1990).

[27] L. Kaufman and P. J. Rousseeuw. “Finding Groups in Data: An Introduction to Cluster Analysis”. In: *John Wiley and Sons* 344 (2009).

[28] J. MacQueen. “Some methods for classification and analysis of multivariate observations”. In: *BSMSP* (1967), pp. 281–297.

[29] L. Kaufman and P. J. Rousseeuw. “Clustering by means of Medoids”. In: *Statistical Data Analysis Based on Norm and Related Methods, edited by Y. Dodge, North-Holland* (1987), pp. 405–416.

[30] M. E. Celebi, H. A. Kingravi, and P. A. Vela. “A comparative study of efficient initialization methods for the k-means clustering algorithm”. In: *Expert Syst. Appl* 40.1 (2013), pp. 200–210.

[31] F. Petitjean, A. Ketterlin, and P. Gançarski. “A global averaging method for dynamic time warping, with applications to clustering”. In: *Pattern Recognition* 44.3 (2011), pp. 678–693.

[32] Y. Kang, D. Belusic, and K. Smith-Miles. “Detecting and Classifying Events in Noisy Time Series”. In: *Journal of the Atmospheric Sciences* 71 (2013), pp. 1090–1104.

[33] X. Wang, K. A. Smith, and R. J. Hyndman. “Characteristic based clustering for time series data”. In: *Data Min. Knowl. Discovery* 13 (2006), pp. 335–364.

[34] V. Niennattrakul and C. A. Ratanamahatana. “On clustering multimedia time series data using K-means and dynamic time warping”. In: *International Conference on Multimedia and Ubiquitous Engineering*. 2007, pp. 733–738.

[35] D. T. Anh and L. Thanh. “An efficient implementation of k-means clustering for time series data with DTW distance”. In: *International Journal of Business Intelligence and Data Mining* 10 (Jan. 2015), pp. 213–232. doi: 10.1504/IJBIDM.2015.071311.

[36] T. Liao et al. “Understanding and projecting the battle state”. In: *23rd Army Science Conference*. 2002.

[37] T.W. Liao, C.-F. Ting, and P.-C. Chang. “An adaptive genetic clustering method for exploratory mining of feature vector and time series data”. In: *International Journal of Production Research* 44 (2006), pp. 2731–2748.

[38] V. Hautamaki, P. Nykanen, and P. Franti. “Time-series clustering by approximate prototypes”. In: *19th International Conference on Pattern Recognition*. 2008, pp. 1–4.

[39] Y. Gao et al. “Improved K -medoids algorithm-based clustering analysis for handle driving force in automotive manual sliding door closing process”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* (Aug. 2020), p. 095440702094582. doi: 10.1177/0954407020945827.

[40] Gholamreza Soleimany and Masoud Abessi. “A New Similarity Measure for Time Series Data Mining Based on Longest Common Subsequence”. In: *American Journal of Data Mining and Knowledge Discovery* 4 (Jan. 2019), p. 32. doi: 10.11648/j.ajdmkd.20190401.16.

[41] E. Keogh and C. Ratanamahatana. “Exact indexing of dynamic time warping”. In: *Knowledge and Information Systems* 7 (Jan. 2005), pp. 358–386. doi: 10.1007/s10115-004-0154-9.

[42] H. Thuy, D. T.Anh, and C. Vo. “Efficient segmentation-based methods for anomaly detection in static and streaming time series under dynamic time warping”. In: *Journal of Intelligent Information Systems* (July 2020). doi: 10.1007/s10844-020-00609-6.

[43] C. W. Tan et al. “Time series classification for varying length series”. Preprints and early-stage research may not have been peer reviewed yet. Oct. 2019.

[44] A. D Hoang et al. *The UCR Time Series Classification Archive*. url: https://www.cs.ucr.edu/~eamonn/time_series_data_2018/ (visited on 03/07/2022).

[45] J. Paparrizos and L. Gravano. “K-shape: Efficient and accurate clustering of time series”. In: *ACM SIGMOD International Conference on Management of Data, ACM*. 2015, pp. 1855–1870.

[46] M. Gabr and L. Fatehy. “Time Series Classification”. In: *Journal of Statistics Applications Probability* 2 (July 2013), pp. 123–133. doi: 10.12785/jsap/020205.

[47] Kuiyong Song, Nianbin Wang, and Hongbin Wang. “A Metric Learning-Based Univariate Time Series Classification Method”. In: *Information* 11 (May 2020), p. 288. doi: 10.3390/info11060288.

[48] Dominique Gay et al. “Multivariate Time Series Classification: A Relational Way”. In: Sept. 2020, pp. 316–330. isbn: 978-3-030-59064-2. doi: 10.1007/978-3-030-59065-9_25.

[49] Feng Chen and Yi-Ping Phoebe Chen. “K-Nearest Neighbor Classification (Lazy Learner, KNN, Instance-based Learner)”. In: Oct. 2004. isbn: 9780471650126. doi: 10.1002/9780471650126.dob0963.

[50] Sachin Gavankar and Sudhirkumar Sawarkar. “Eager decision tree”. In: Apr. 2017, pp. 837–840. doi: 10.1109/I2CT.2017.8226246.

[51] Davide Anguita et al. “The ‘K’ in K-fold Cross Validation”. In: Jan. 2012.

[52] Tuan Tran et al. “A novel non-parametric method for time series classification based on k -Nearest Neighbors and Dynamic Time Warping Barycenter Averaging”. In: *Engineering Applications of Artificial Intelligence* 78 (Feb. 2019), pp. 173–185. doi: 10.1016/j.engappai.2018.11.009.

[53] Houtao Deng et al. “A Time Series Forest for Classification and Feature Extraction”. In: *Information Sciences* 239 (Aug. 2013), pp. 142–153. doi: 10.1016/j.ins.2013.02.030.

[54] Michael Flynn, James Large, and Tony Bagnall. “The Contract Random Interval Spectral Ensemble (c-RISE): The Effect of Contracting a Classifier on Accuracy”. In: Aug. 2019, pp. 381–392. isbn: 978-3-030-29858-6. doi: 10.1007/978-3-030-29859-3_33.

[55] Jitao Zhang et al. “Time Series Classification by Shapelet Dictionary Learning with SVM-Based Ensemble Classifier”. In: *Computational Intelligence and Neuroscience* 2021 (Mar. 2021), pp. 1–13. doi: 10.1155/2021/5586273.

[56] Amaia Abanda, Usue Mori, and Jose Lozano. “A review on distance based time series classification”. In: *Data Mining and Knowledge Discovery* 33 (Mar. 2019). doi: 10.1007/s10618-018-0596-4.

[57] N Nirsal et al. “Classification of medicine characteristic using Super Vector Machine (SVM) at Palopo regional public Hospital Sawerigading”. In: *Journal of Physics: Conference Series* 1833 (Mar. 2021), p. 012028. doi: 10.1088/1742-6596/1833/1/012028.

[58] Zheng zheng Xing, Jian Pei, and Eamonn Keogh. “A Brief Survey on Sequence Classification”. In: *SIGKDD Explorations* 12 (Nov. 2010), pp. 40–48. doi: 10.1145/1882471.1882478.

[59] Christos Faloutsos, M. Ranganathan, and Yannis Manolopoulos. “Fast Subsequence Matching in Time-Series Databases”. In: *ACM SIGMOD Record* 23 (June 2000). doi: 10.1145/191839.191925.

[60] Flip Korn, H. Jagadish, and Christos Faloutsos. “Efficiently Supporting Ad Hoc Queries in Large Datasets of Time Sequences”. In: *SIGMOD Record (ACM Special Interest Group on Management of Data)* 26 (Apr. 1999). doi: 10.1145/253262.253332.

[61] Anthony Bagnall. “A Run Length Transformation for Discriminating Between Auto Regressive Time Series”. In: *Journal of Classification* 31 (July 2013). doi: 10.1007/s00357-013-9135-6.

[62] Marcella Corduas and Domenico Piccolo. “Time series clustering and classification by the autoregressive metric”. In: *Computational Statistics Data Analysis* 52 (Feb. 2008), pp. 1860–1872. doi: 10.1016/j.csda.2007.06.001.

[63] Padhraic Smyth. “Clustering Sequences with Hidden Markov Models”. In: *Advances in Neural Information Processing Systems* 9 (July 1999).

[64] The New York Times. *Russia Orders Evacuation of Village Near Site of Nuclear Explosion*. url: https://www.nytimes.com/2019/08/13/world/europe/russia-nuclear-explosion-accident.html (visited on 03/07/2022).

[65] Tan et al. *Introduction to Data Mining*. May 2005.

[66] Jiawei Han and Micheline Kamber. *Data Mining Concepts and Techniques (2nd Edition)*. Jan. 2006. isbn: 1-55860-901-6.

[67] Prajak Barde and Prajakt Barde. “What to use to express the variability of data: Standard deviation or standard error of mean?” In: *Perspectives in clinical research* 3 (July 2012), pp. 113–6. doi: 10.4103/2229-3485.100662.