Ref. Ares(2021)7311105 - 27/11/2021

H2020 – ICT-13-2018-2019

MUSKETEER

Machine Learning to Augment Shared Knowledge in Federated Privacy-Preserving Scenarios (MUSKETEER)

Grant No 824988

D7.6 Use case execution and KPI evaluation in the Health domain

November 21

Imprint

Contractual Date of Delivery to the EC: **30 November 2021**

Author(s): **Susanna Bonura (ENG), Joao Correia (B3D), Walter Hernndez (B3D), Petros Papachristou (HYGEIA), Christina Kotsiopoulou (HYGEIA)**

Participant(s): **Davide Profeta (ENG), Jaime Medina (TREE), Mark Purcell (IBM), Angel Navia Vazquez (UC3M)**

Reviewer(s): **Lucrezia Morabito (COMAU), Antoine Garnier (IDS)**

Project: **Machine learning to augment shared knowledge in federated privacy-preserving scenarios (MUSKETEER)**

Work package: **WP7**
Dissemination level: **Public**
Version: **1.0**

Contact: **jcorreia (at) biotronics3d.com**
Website: **www.MUSKETEER.eu**

Legal disclaimer

The project Machine Learning to Augment Shared Knowledge in Federated Privacy-Preserving Scenarios (MUSKETEER) has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824988. The sole responsibility for the content of this publication lies with the authors.

Copyright

Executive Summary

This deliverable D7.6 Use case execution and KPI evaluation in the Health domain, presents the results of the work carried out in the tasks T7.3 Use cases execution, and T7.4 Use cases validation and KPI evaluation, in the Health domain. Throughout these tasks, the partners deployed the MUSKETEER platform components for Federated Machine Learning on two pilot sites, prepared and pre-processed data, trained FML models and evaluated the results.

The document presents the technical and business perspectives of the evaluation of MUSKETEER platform, and the business general performance indicators for the health pilot. It provides details of the health use case, describing the context of the pilot, the data sets, the pre-processing modules implemented and MUSKETEER components used, the setup and execution of the health pilot and its results. It reports the evaluation of the KPIs in order to assess the usage of the MUSKETEER platform on the health pilot, from a business and technical perspective, as defined in the deliverable D2.7 - Key performance indicators selection and definition. Finally, the deliverable presents the conclusions of the health use case execution, validation and KPI evaluation performed on tasks T7.3 and T7.4.

Document History

Version	Date	Status	Author	Comment
0.1	21/09/2021	Structure	Susanna Bonura	
0.2	22/09/2021	Initial content	Joao Correia	
0.3	30/09/2021	Use case, POM	Joao Correia	
0.4	15/10/2021	Pre-processing, ML, datasets, Pilot setup and execution	Walter Hernandez, Christina Kotsiopoulou	
0.5	29/10/2021	KPIs and Business metrics	Joao Correia, Walter Hernandez, Petros Papachristou, Christina Kotsiopoulou	
0.6	03/11/2021	Technical metrics	Susanna Bonura, Jaime Medina, Mark Purcell, Angel Navia Vazquez	
0.7	05/11/2021	Results	Walter Hernandez	
0.8	15/11/2021	Update results and evaluation, conclusion	Joao Correia, Walter Hernandez, Petros Papachristou, Christina Kotsiopoulou	
0.9	18/11/2021	Reviewers' comments	Lucrezia Morabito, Antoine Garnier	
1.0	26/11/2021	Final version	Joao Correia, Gal Weiss, Mark Purcell	

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Abbreviation	Definition
API	Application Programming Interface
CA	Consortium Agreement
CC	Client Connector
CNN	Convolutional Neural Network
DP	Differential Privacy
DC	Data Connector
DICOM	Digital Imaging and Communications in Medicine
DV	Data Value
FS	Feature Selection
FSM	Finite State Machine
GA	Grant Agreement
IDR	Intermediate Data Representation
IDS	Industrial Data Space
LC	Logistic Classifier
LGFS	Linear Greedy Feature Selection
MK	Master Key
ML	Machine Learning
MLP	Multi-Layer Perceptron
MN	Master Node
MRI	Magnetic resonance imaging
OS	Operating System
PACS	Picture archiving and communication system
PERT	Program evaluation and review technique
PI-RADS	Prostate Imaging–Reporting and Data System – Structured reporting with scoring system
PK	Public Key
POM	Privacy Operation Mode
PP	Privacy Preserving
PPML	Privacy Preserving Machine Learning
RAM	Reference Architecture Model
ROC	Receiver Operating Characteristics
SQL	Structured Query Language
TA	Task Alignment
UI	User Interface
WN	Worker Node

1 Introduction

1.1 Purpose

This document, resulting from tasks T7.3 and T7.4, describes the setup and execution of the MUSKETEER Health pilot, and reports the evaluation of the KPIs in order to assess the usage of the platform in a real-world use case scenario.

On task T7.3 the partners performed the setup of the components that enable the interaction with the MUSKETEER Federated Machine Learning platform services for the health pilot at Biotronics3D and Hygeia premises. Data anonymisation and transformation were implemented in pre-processing modules to enable the adequate use of health imaging data in the MUSKETEER platform. The pilot environment was properly deployed at B3D and Hygeia in order to obtain a fully working test bed. The health use case scenario has been executed iteratively to collect information regarding the experience of the demonstrator partners with the MUSKETEER platform.

On task T7.4, the evaluation of the MUSKETEER platform though the health pilot was carried out, focusing on the correct adoption of the platform including benchmarking and impact assessment based on the requirements. The KPI checklist was used to validate the platform performance. The KPI evaluation enabled to identify the necessary adjustments to align the results to the original requirements.

1.2 Related Documents

The documents related with this deliverable are:

D2.1 – Industrial and technical requirements

D2.3 – Key performance indicators selection and definition

D2.4 - Privacy and confidentiality impact assessment report and recommendations – Initial version

D2.5 - Privacy and confidentiality impact assessment report and recommendations – Final version

D2.7 – Key performance indicators selection and definition - final version

D3.2 Architecture Design – Final Version

D7.1 – Client connectors’ architecture design – Initial version

D7.2 – Client connectors architecture design – Final version

D7.3 – First prototype of the MUSKETEER client connectors

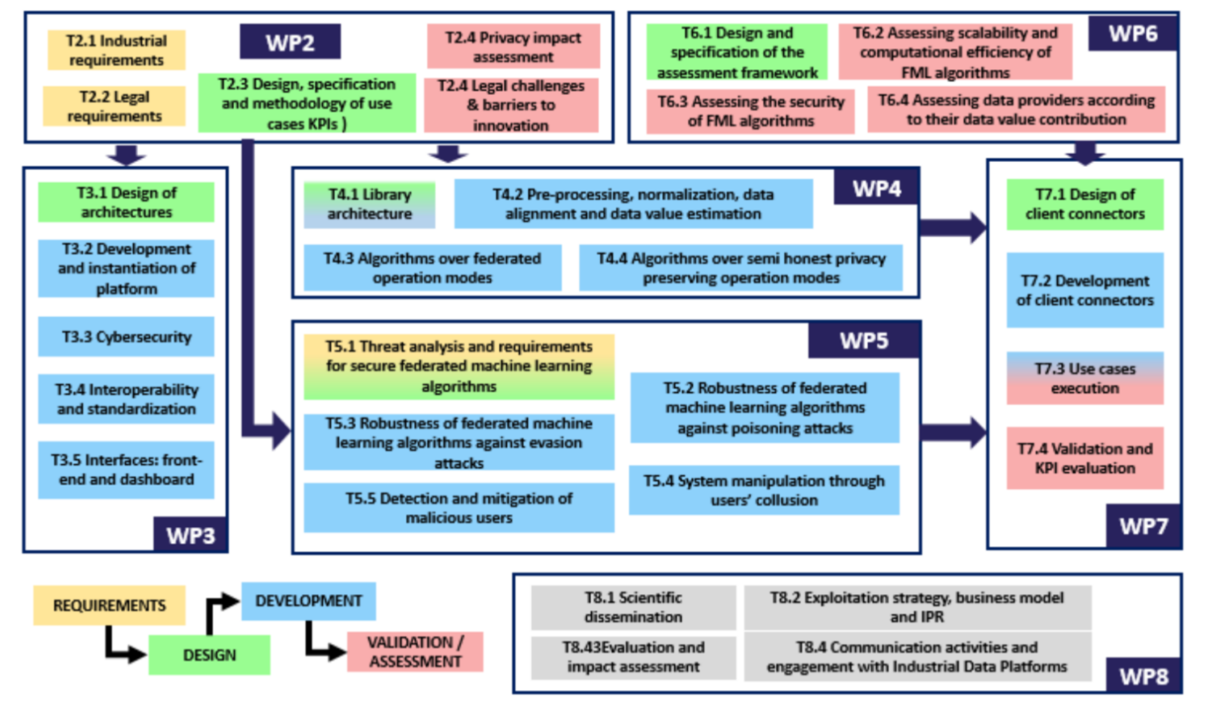

Figure 1 - MUSKETEER's PERT diagram

1.3 Document Structure

The document starts with this introductory chapter that presents its purpose, related documents and the structure of the deliverable.

The technical and business perspectives of the evaluation of MUSKETEER platform are presented on chapter two.

Chapter three reports the business general performance indicators for the health pilot.

On chapter four is provided a detailed presentation of the health use case, describing the context of the pilot, the data sets, the pre-processing modules implemented and MUSKETEER components used, the setup and execution of the health pilot and its results.

Chapters five and six, report the evaluation of the KPIs in order to assess the usage of the MUSKETEER platform on the health pilot, from a business and technical perspective, as defined in the final version of the deliverable D2.7 - Key performance indicators selection and definition.

The last chapter presents the conclusions of the health use case execution, validation and KPI evaluation performed on tasks T7.3 and T7.4.

2 MUSKETEER Evaluation perspectives

The MUSKETEER Evaluation Framework contains four phases inspired by the GQM method (cf. D2.7 for further details):

- Planning phase, executed within WP2.
- Definition phase, executed within WP2.
- Data Collection phase, addressed within WP7 and described in this document.
- Interpretation phase, addressed within WP7 and described in this document.

In accordance with the MUSKETEER Evaluation Framework for validating the MUSKETEER platform, each evaluation object defined for the health use case will be validated according to two different perspectives that are defined below:

- The Technical Perspective, in which some aspects of the MUSKETEER architecture, Privacy preserving FML algorithms under different POMs and FML Algorithms will be evaluated.
- The Business Perspective, in which the response to the user will be examined. The instrumentation used in this perspective is drawn upon usefulness and feasibility of the proposed deployment operation support.

3 Business General Performance Indicators

Quantitative and qualitative evaluation metrics that correspond to the evaluation of the MUSKETEER platform operation phase for product validation purposes are presented in the table below. Many of the general performance indicators that are adopted refer to the ISO/IEC 25010:2011 standard [3] for usability aspects, and are measured in a qualitative manner, either by measuring AS-IS and TO-BE values, or in case of more qualitative answers, by using a 1-5 scale. In the last column, we show the evaluation done by the end-users Hygeia and B3D.

Characteristics	**Identifier**	**Metric**	**Definition**	**Mandatory (Y/N)**	**Evaluation**
	Business Value				
Clarity	GPI_BusV1	Clarity level	How clear was it for you what the MUSKETEER platform is about? [Scale 1 (Little) -5 (Very)]	Y	5

Added Value	GPI_ BusV2	Added value level	How much added value do you feel that the MUSKETEER platform provides to your operations while using it? [Scale 1 (Low) -5 (High)]	Y	4
Need Importance Level	GPI_ BusV3	Need importance level	How important is for you the need that the MUSKETEER platform covers for you? [Scale 1 (Little) -5 (Very)]	Y	5
Need Coverage	GPI_ BusV4	Need coverage level	To which degree does the MUSKETEER platform cover your needs? [Scale 1 (Low) -5 (High)]	Y	4
Innovation	GPI_ BusV5	Innovation level	How innovative do you find the idea of the MUSKETEER platform? [Scale 1 (Little) -5 (Very)]	Y	5
Intention to use	GPI_ BusV6	Intention level	To what extent do you intend to use the MUSKETEER platform? [Scale 1 (Low) - 5 (High)]	N	5
Virality	GPI_ BusV7	Virality level	How probable is it for you to recommend the MUSKETEER platform to other stakeholders? [Scale 1 (Low) -5 (High)]	Y	5
	Effectiveness				
Effectiveness	GPI_Effe1	Effectiveness level	Is the MUSKETEER platform enabling you to accurately achieve your goals for data sharing and data analytics? [Scale 1 (Low) -5 (High)]	Y	4
	Efficiency				
Efficiency	GPI_Effi1	Efficiency level	Is the MUSKETEER platform efficiently fulfilling its intended	Y	4

			purpose? [Scale 1 (Low) -5 (High)]		
	Satisfaction				
Usefulness	GPI_Sati1	Usefulness level	Do you find the MUSKETEER platform useful? [Scale 1 (Low) -5 (High)]	Y	4
Trust	GPI_Sati2	Trust level	Do you trust the MUSKETEER platform and its provided functionalities? [Scale 1 (Low) -5 (High)]	Y	4
Pleasure	GPI_Sati3	Pleasure level	Does the MUSKETEER platform please you when you use it? [Scale 1 (Low) -5 (High)]	Y	4
Comfort	GPI_Sati4	Comfort level	Do you feel that the MUSKETEER platform provides a comfortable user interface and workflows? [Scale 1 (Low) -5 (High)]	Y	4
	Freedom from risk				
Economic damage risk	GPI_Free1	Level of economic damage risk	How sure are you that MUSKETEER protects you from exposing you on economic damage? [Scale 1 (Low) -5 (High)]	Y	5
Privacy harm risk	GPI_Free2	Level of data privacy damage risk	How sure are you that MUSKETEER is on protecting your data privacy? [Scale 1 (Low) -5 (High)]	Y	5
	Usability				
Learnability	GPI_Usab1	Learnability level	How easy it was for you to learn how to use the MUSKETEER platform? [Scale 1 (Not) -5 (Very)]	Y	3
Flexibility	GPI_Usab2	Flexibility level	How much do you believe the MUSKETEER platform can be used for other applications than the demonstrator ones?	Y	4

			[Scale 1 (Low) -5 (High)]		
	Content Conformity				
Quality	GPI_Conte1	Content quality	How useful do you find the data and the applications found in the MUSKETEER platform in terms of quality? [Scale 1 (Little) -5 (Very)]	Y	4
Quantity	GPI_Conte2	Content quantity	How satisfied are you from the quantity of the data and the algorithms found in the MUSKETEER platform? [Scale 1 (Little)-5 (Very)]	Y	4

4 Health use case

Health data is a very special type of personal data that encompasses an extreme value for the patient, the data subject, and for the healthcare practitioners who should decide on the correct diagnosis and care pathways to achieve the best patient outcomes. Health data is also extremely important to the research, development and validation of new technologies, procedures and care pathways to improve the diagnosis, prognosis and treatment of diseases.

The recent years have shown important advances in Artificial Intelligence, enabled by cloud-computing and big-data collections, with application in many different fields, including the health care sector. One key element for improving AI algorithms and its results is gathering large amounts of good-quality data. In the health care sector, mainly for security and privacy reasons, but also due to some lack of interoperability and standardisation, it has been difficult to concentrate large amounts of quality data for the development of AI methodologies. Biobanks are vital source of information for fundamental and translational biomedical research aimed at the development of better predictive, preventive, personalised and participatory health care. Although a large majority of world biobanks are located in Europe, until recently, imaging data coming from magnetic resonance imaging (MRI) or computed tomography (CT) were not included in such biobanks. Projects have been launched to acquire large repositories of image data, but in 80% of cases the access to imaging biobanks is restricted to research and clinical reference. Multi-tenant and multi-datacentre cloud solutions for medical imaging management, analysis and reporting, have been used in clinical practice for radiology and tele-radiology for a few years. They have been used by public hospitals to organise networked, collaborative reporting services, and by private practices to improve the productivity on large distributed groups and on small clinics. Vast amounts of medical imaging data are collected and reported using these cloud solutions, but each organisation accesses only its own data. Thanks to MUSKETEER the limitation of accessing large amounts of data is being surpassed.

The pressure for productivity is increasing due to the lack of Radiologists and the growing demand for medical imaging services. Key driving factors are the rise in prevalence of chronic diseases, technological advancements in diagnostic imaging modalities, increasing number of imaging procedures, rising awareness among the patients about early diagnosis of clinical disorders and rise in base of aging population. In addition, increasing demand from emerging countries, improved government funding towards chronic disorders, increasing investment in public and private organizations, and increasing disposable income among the population will further expected to drive the market in the coming years.

The European Union, the US and many other countries have been focusing their public health policies and research efforts on personalised medicine and evidence based clinical pathways to improve patient outcomes and effectiveness of care. The solution lies in providing powerful tools to support the radiologists to take faster and more accurate decisions for diagnosis and prognosis. As some research projects have been indicating, AI and Machine Learning (ML) are disruptive technologies that will enable develop powerful tools leading to improvements to the clinical protocol pathways and conducting to better efficiencies and better patients' outcomes.

This use case intends to demonstrate the application of the MUSKETEER Federated Machine Learning platform, enabling access to vast amounts of distributed medical imaging data to train and improve the learning algorithms, in a federated privacy-preserving environment. The main objective is training AI/ML algorithms for support the detection of prostate cancer. Since it is really hard to collect medical records, the benefit to collaborate providing datasets to improve the learning of predictive models to support the medical diagnosis is clear.

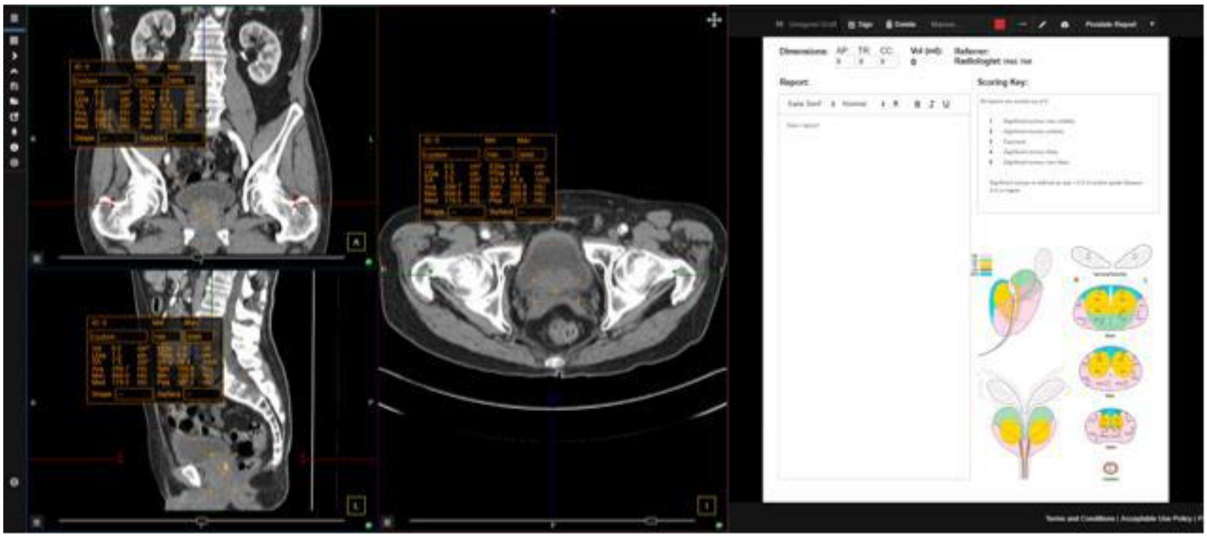

Figure 2. Prostate image visualisation and report

B3D and HYGEIA can take a huge advantage of MUSKETEER developments to demonstrate the application of the Artificial Intelligence methodologies and technologies enabling access to distributed medical imaging data to train and improve the learning algorithms, providing powerful tools to improve clinical practice. The expected impacts of this use case are several:

1. Improve accuracy of AI algorithms by sharing knowledge from distinct organisations and data repositories, supporting cooperation keeping security and privacy of health data;
2. More accurate clinical decision support tools for diagnosis and prognosis of diseases, avoiding invasive procedures and conducting to better patient outcomes;
3. Faster decision support tools, enabling shorter turn-around-times, increasing productivity of services and more studies and patients diagnosed;
4. Faster and more accurate clinical decision support tools for diagnosis and prognosis saving lives in emergency cases;
5. Enable the growth of the level of research in medical imaging AI tools supported by distributed data repositories;
6. Enable clinical practices to access medical imaging AI tools with gains of productivity and better patient outcomes;
7. Improve Biotronics3D commercial offer, enabling partners to access its market.

4.1 Pre-processing, ML Algorithms and POM

4.1.1 Pre-processing description

Clinical data at Biotronics3D UK's site was chosen from a pre-existing metadata file that contained information about lesion classification and location in the prostate. Each imaging study was comprised of series with an average of 20 slices of 385x385 pixels.

Clinical data at Hygeia Greece's site was chosen from a set of annotations recently performed by medical doctors using the 3Dnet system. The database storing these annotations was then queried in order to obtain the information about lesions classification and location in the prostate. Each imaging study was comprised of series with an average of 20 slices of different dimensions 116x116, 640x640 and 700x640 pixels.

Studies with lesions tagged as "clinically significant" or "malignant" and "non-clinically significant" or "benign" were anonymised and copied in different spaces in order to generate the required labels for the models. For the 2D models, images were resized in both sites to 128x128 and for the 3D models to 64x64x16. Pixel integer values (16 bits) were converted to float and normalized to the range 0-1.

Data was augmented in order to complement a low number of cases and also to find a balance in the number of slices (2D data) and volumes (3D data) between sites. The data augmentation technique involved a random variance in the brightness of (+/- 0.1 to 0.5 of a normalized pixel/voxel value) and rotation (+/- 0 to 5 degrees around the axial direction) which are typical variances shown among acquisition equipment.

4.1.2 ML algorithms

2D CNN model using the Client Connector (CC) for classification of Clinically Significant lesions:

- Significant slices from the volume are treated individually and radiologists' labels are extracted as True ("clinically significant") or False ("non-clinically significant")
- All slices are reshaped to 128x128, as illustrated in **Figure 3**, and converted to the pickle format supported by the CC.
- CNN model uses three 2D convolution layers with 100 iterations.
- Model aims to determine if a slice contains a clinically significant lesion

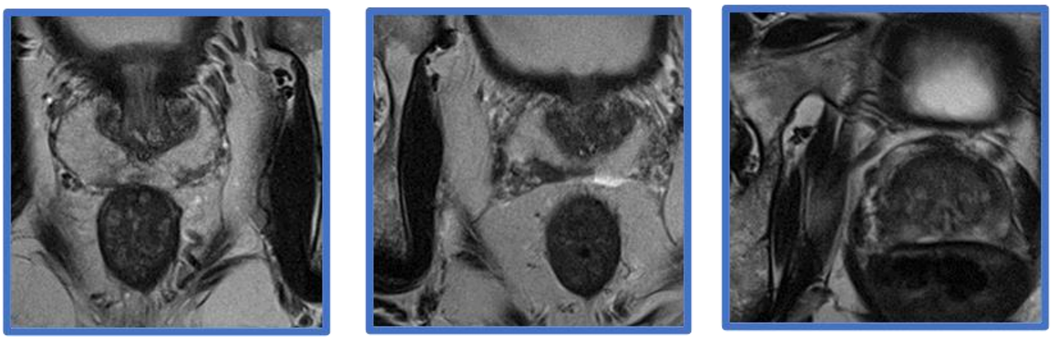

Figure 3. 2D images used in 2D CNN model training

3D CNN model using the Client Connector (CC) for classification of Clinically Significant lesions:

- Full volume is classified as "clinical significant" if it contains at least one clinically significant lesion and "non-clinical significant" otherwise, according to radiologists' labels
- All volumes are reshaped to 64x64x16, as illustrated in **Figure 4**, and converted to the pickle format supported by the CC.
- CNN model uses three 3D convolution layers with 100 iterations.

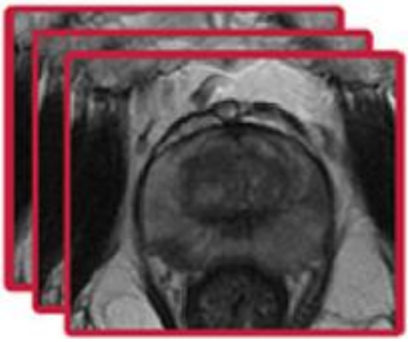

Figure 4. 3D images used in 3D CNN model training

The 2D and 3D CNN models are detailed in Annex A – 2D and 3D CNN models.

4.1.3 POM

Machine learning algorithms can process health records to create predictive models capable to help in the medical diagnosis, these types of datasets are very valuable for research purposes. For a single hospital it is very complicate to collect a dataset large enough to create a complex predictive model. For that reason, the benefit, in terms of predictive model accuracy, of combining datasets of different hospitals is very clear. However, having explicit consent of a patient to use his/her health records does not guarantee the protection of security and privacy and when two or more different research groups have explicit permission to use a health dataset, different barriers arise.

The EU General Data Protection Regulation EU 2016/679 (GDPR) defines health records as special categories of personal data, which have very restrictive processing. The MUSKETEER partners conducted a Data Protection Impact Assessment to verify the adequacy of the processing considering the implemented pre-processing methodologies before any data is made available to the MUSKETEER platform. The pre-processing methodologies have two main steps: use DICOM anonymiser tools to remove any personal identifiable data from the imaging headers; and vectorise the data by transforming the pixel data, measurements and labels into sequences of numbers that do not contain any identifier that can be related to the patient. Therefore, the data that is made accessible to the MUSKETEER platform is anonymous data, out of the scope of GDPR. This transformation pre-processing is done at the premises of the partners where the PACS system resides. Moreover, the MUSKETEER Client Connector, the only component that has access to the provided anonymous data, is also installed at the infrastructure of each partner providing data. The only data transferred between the MUSKETEER Client Connector and the MUSKETEER Server are coefficients and gradients of the AI/ML models trained locally and averaged at the Server node.

The partners involved in the Health pilot have considered the different POMs available and have selected at the beginning of the project the POMs PORTHOS and RICHELIEU. However, at the pilot implementation stage, having considered the requirements of each POM and the effective nature of data transferred, which is previously anonymised and vectorised as presented on the previous paragraph, the partners decided to use POM 1 Aramis.

4.2 Data set

Biotronics3D is using publicly available datasets curated by The Cancer Imaging Archive (TCIA). Hygeia uses private datasets from their hospital which are anonymised before being pre-processed for use in MUSKETEER.

The following table presents the imaging datasets of relevant cases that were selected during the project from original datasets to use in the Health pilot.

Table 1. Characteristics of original datasets provided by Hygeia and B3D (selection phase)

Dataset	Available at	#Exams	T1	T2	PI-RADS/ Clinical Significance

Multi-parametric MRI of Prostate Gland	Hygeia	47	47	47	45
TCIA PROSTATEx	B3D	100	-	100	100

TCIA Data Citation: Geert Litjens, Oscar Debats, Jelle Barentsz, Nico Karssemeijer, and Henkjan Huisman. "ProstateX Challenge data", The Cancer Imaging Archive (2017). DOI: 10.7937/K9TCIA.2017.MURS5CL

Table 2. Characteristics of final datasets used on the 2D CNN tests

Dataset name	Original (slices)	Augmented (slices)
uk_tra_2d.pkl	70	280
uk_val_2d.pkl	30	120
uk_tes_2d.pkl	50	200
gr_tra_2d.pkl	29	257
gr_val_2d.pkl	9	105
gr_tes_2d.pkl	9	105

Table 3. Characteristics of final datasets used on the 3D CNN tests

Dataset name	Original (volumes)	Augmented (volumes)
uk_tra_3d.pkl	90	n/a
uk_val_3d.pkl	25	n/a
uk_tes_3d.pkl	25	n/a
gr_tra_3d.pkl	26	100
gr_val_3d.pkl	8	35
gr_tes_3d.pkl	8	35

Table 4. Sample header of 2D vector file after pre-processing step

Field(s)	Description	# Features	Labels
l_Nlesions	Number of lesions found (number)	1	
l_LesionPresence1	Lesion 1 Presence (Y/N)	1	

I_LesionPresence2	Lesion 2 Presence (Y/N)	1	
I_T2S01P0101 to I_T2S01P128128	Pixels from T2 Slices (128x128)	16.384	
Malignant	Lesion present is Benign or Malignant		1

Table 5. Sample header of 3D vector file after pre-processing step

Field(s)	**Description**	**# Features**	**Labels**
I_Nlesions	Number of lesions found (number)	1	
I_LesionPresence1	Lesion 1 Presence (Y/N)	1	
I_LesionPresence2	Lesion 2 Presence (Y/N)	1	
I_T2V01V0101 to I_T2V16V6464	Voxels from T2 Volumes (16x64x64)	65.536	
Malignant	Lesion present is Benign or Malignant		1

4.3 Pilot setup and execution

The setup of the MUSKETEER components for the health pilot has been performed by Biotronics3D and Hygeia with the support of Engineering, IBM and TREE. Biotronics3D and Hygeia use 3Dnet PACS and Advanced Viewer for medical imaging, which is developed by Biotronics3D, to visualize communicate and archive imaging studies. Biotronics3D developed a special pre-processing module, as an extension to 3Dnet, to transform the selected studies into anonymized vector data. Biotronics3D and Hygeia installed the MUSKETEER Client Connector in specific virtual machines inside their infrastructure. The MUSKETEER Client Connector operates with the MUSKETEER server provided by IBM.

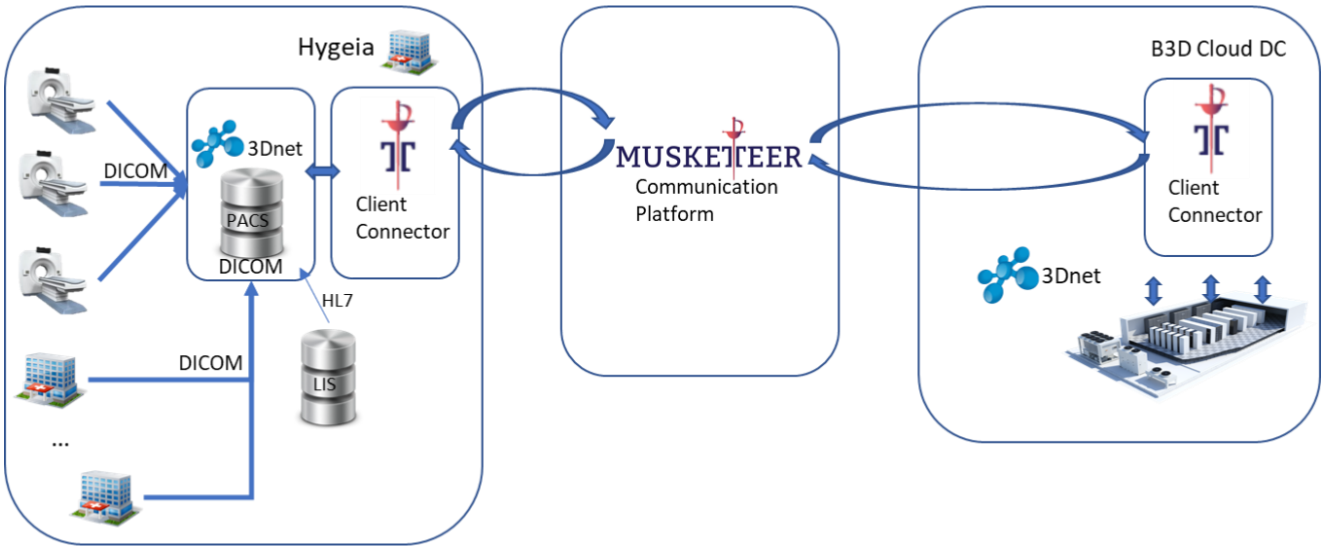

Figure 5. Setup of MUSKETEER components for the Health pilot

Inside the premises of each partner running the health pilot there are three main components: 3Dnet with a specific pre-processing module to transform the selected studies into anonymized vector data; the shared folder that allows to provide datasets to the Client Connector and receive the resulting ML models; and the MUSKETEER Client Connector that allows to create tasks for training ML models, interacting with the MUSKETEER Server to improve the models and return the results.

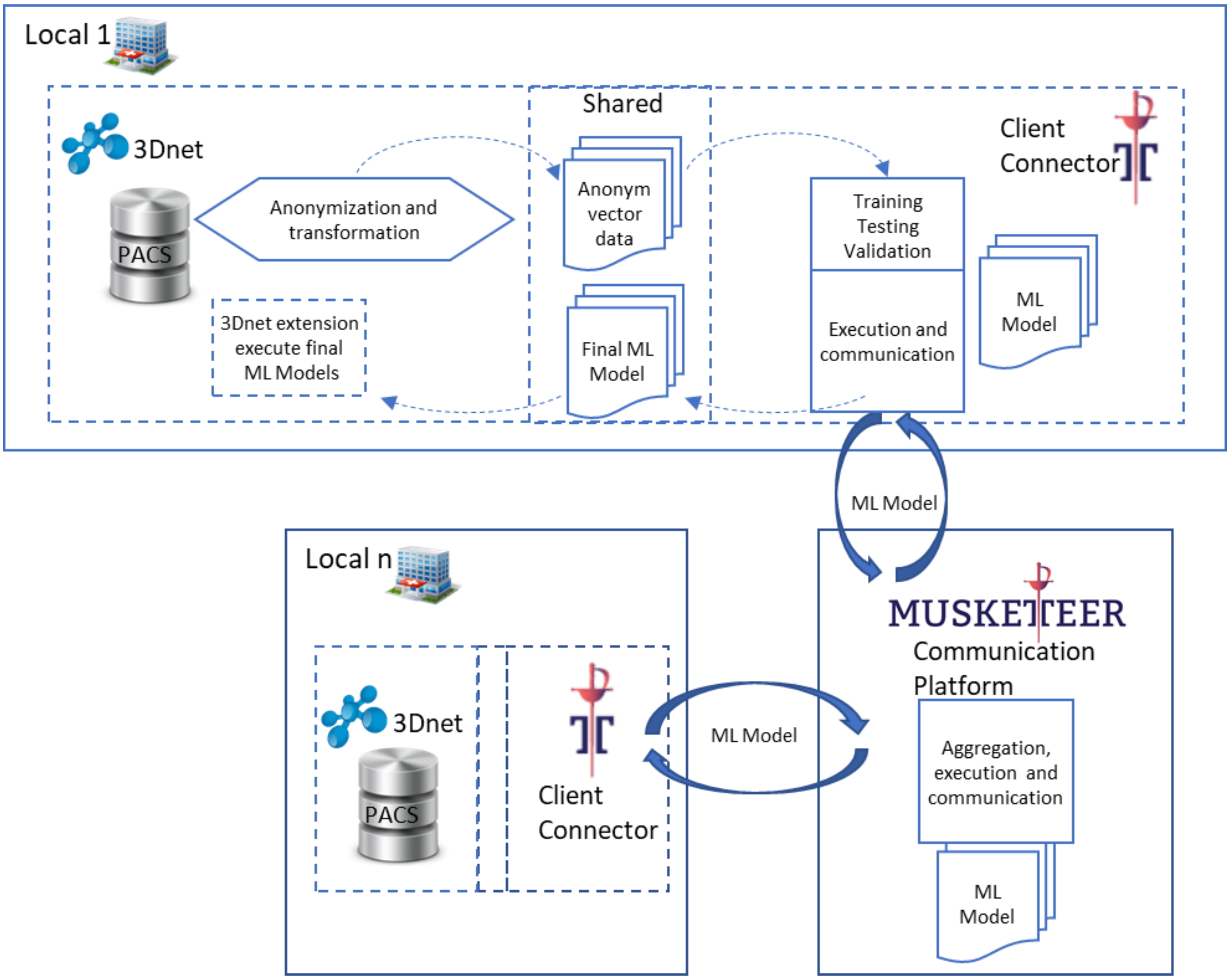

Figure 6. Data flow in the MUSKETEER Health pilot

No personal identifiable information is passed to the MUSKETEER platform. Only anonymized vector data is shared with the MUSKETEER Client Connector, inside the premises of the partners, and only model information and coefficients are shared with the MUSKETEER Server outside the premises of the partners.

The two CNN models were trained in the MUSKETEER Federated Machine Learning platform.

The 2D and 3D datasets from B3D and HYGEIA were divided in training, test and validation as shown in **Figure 7** and **Figure 8**.

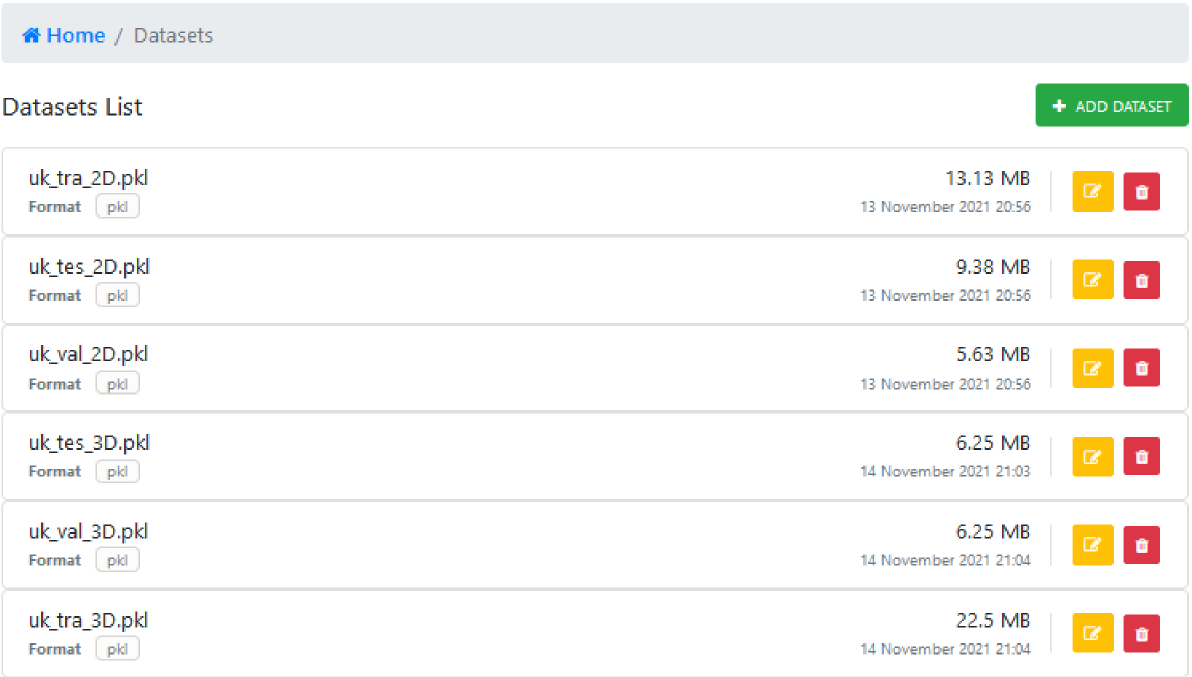

Figure 7. B3D 2D and 3D Datasets

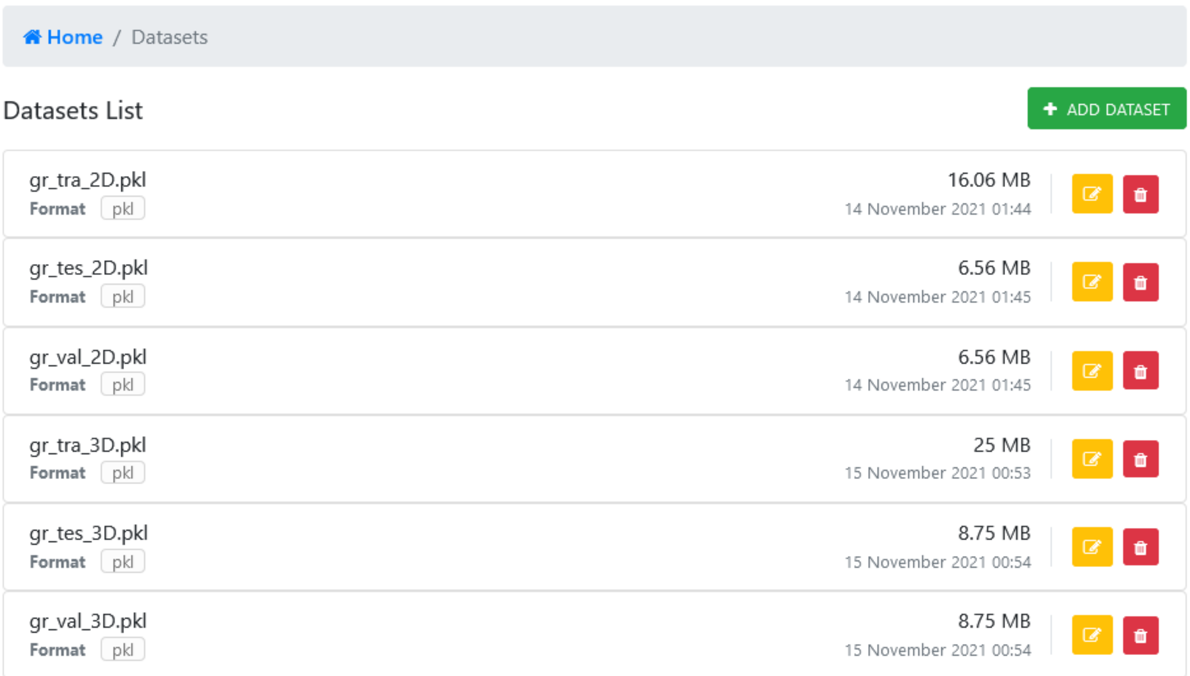

Figure 8. HYGEIA 2D and 3D Datasets

The aggregator created the 2D CNN task and defined the parameters as presented in **Figure 9**.

Data Description

Features	16384
Labels	2
Input Data Description	Details

Preprocessing

Advanced Information

Nmaxiter	100
batch_size	16
learning_rate	0.0015
loss	binary_crossentropy
metric	accuracy
model_averaging	False
momentum	1
nesterov	false
num_epochs	50
optimizer	SGD

Figure 9. Configuration of the 2D CNN task

The aggregator created the 3D CNN task and defined the parameters as presented in **Figure 10**.

Data Description

Features	65536
Labels	2
Input Data Description	Details

Preprocessing

Advanced Information

Nmaxiter	100
batch_size	16
learning_rate	0.0015
loss	binary_crossentropy
metric	accuracy
model_averaging	False
momentum	0.9
nesterov	false
num_epochs	50
optimizer	SGD

Figure 10. Configuration of the 3D CNN task

4.4 Results

In order to verify the advantages of using Federated Machine Learning we ran 3 tests for each model using only B3D data, only Hygeia data and both sites' data on the MUSKETEER FML environment.

4.4.1 2D CNN Results

TEST 1: only B3D 2D data

<table>
<tr><th colspan="6">Task definition</th></tr>
<tr><td>Task Name</td><td colspan="5">b3d_2D_u_06</td></tr>
<tr><td>Description</td><td colspan="5">Training 2D CNN federated model to classify prostate cancer images</td></tr>
<tr><td>POM</td><td>POM1</td><td>Algorithm</td><td>CNN</td><td>Quorum</td><td>1</td></tr>
<tr><td>Max Iterations</td><td>100</td><td>Pre-processing</td><td colspan="3">None in CC</td></tr>
</table>

<table>
<tr><th colspan="5"><u>Task execution</u></th></tr>
<tr><td rowspan="3">Aggregator</td><td>User</td><td colspan="3">b3d_node1</td></tr>
<tr><td>Val dataset</td><td colspan="3">uk_val_2D.pkl</td></tr>
<tr><td>Test dataset</td><td colspan="3">uk_tes_2D.pkl</td></tr>
<tr><td rowspan="4">Participants</td><td rowspan="4">User</td><td colspan="3">b3d_node2</td></tr>
<tr><td>Train dataset</td><td colspan="2">uk_tra_2D.pkl</td></tr>
<tr><td>Val dataset</td><td colspan="2">uk_val_2D.pkl</td></tr>
<tr><td>Test dataset</td><td colspan="2">uk_tes_2D.pkl</td></tr>
</table>

The accuracy of the task, made available by the platform on the participant side, is reported in the following table.

<u>Task Results</u>	
Accuracy b3d_node2 (Non-federated/Local)	75.71%

The task result is also presented on the MUSKETEER Client Connector interface as a confusion matrix. This is a specific table layout that allows visualization of the performance of an algorithm. On the horizontal axis there are predicted labels and on the vertical axis the true labels. On each cell (x,y) is reported the percentage of instances predicted by the algorithm with label x that really have label y. The higher are the values on the diagonal, the higher is the quality of the prediction. Notice that, this matrix is calculated using the data available in the test dataset on participant side.

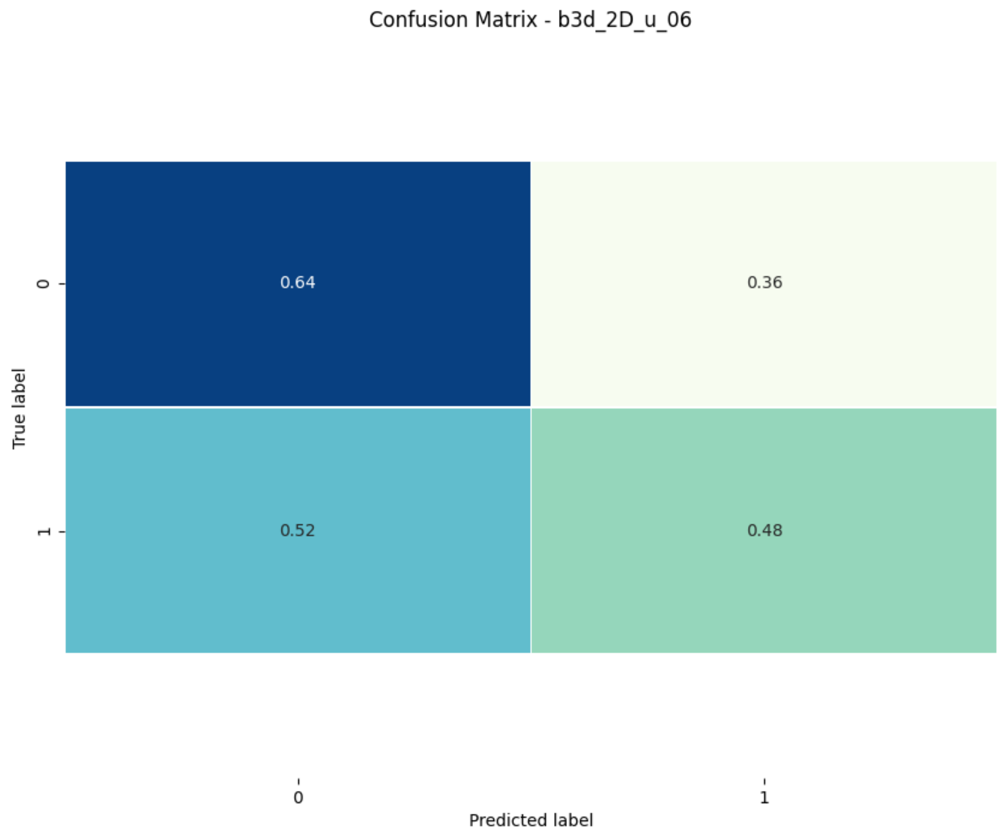

Figure 11. Confusion Matrix for 2D CNN B3D training

TEST 2: only Hygeia 2D data

Task definition					
Task Name	b3d_2D_g_07				
Description	Training 2D CNN federated model to classify prostate cancer images				
POM	POM1	**Algorithm**	CNN	**Quorum**	1
Max Iterations	100	**Pre-processing**	None in CC		
Task execution					
Aggregator	**User**	b3d_node2			
	Val dataset	gr_val_2D.pkl			
	Test dataset	gr_tes_2D.pkl			
Participants	**User**	b3d_node3			
		Train dataset	gr_tra_2D.pkl		
		Val dataset	gr_val_2D.pkl		
		Test dataset	gr_tes_2D.pkl		

The accuracy of the task is reported in the following table.

Task Results	
Accuracy b3d_node3 (Non-federated/Local)	70.43%

The confusion matrix for this task is presented in **Figure 12**.

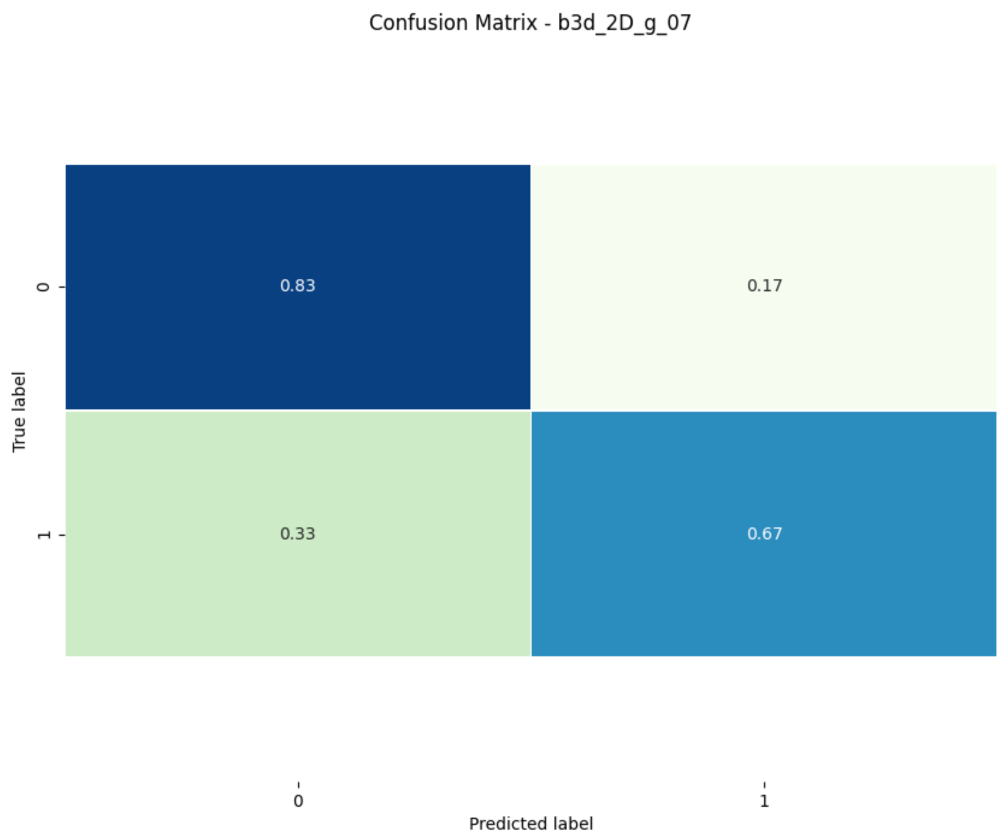

Figure 12. Confusion Matrix for 2D CNN Hygeia training

TEST 3: B3D and Hygeia 2D data

<table>
<tr><th colspan="6"><u>Task definition</u></th></tr>
<tr><td>Task Name</td><td colspan="5">b3d_2D_ug_01</td></tr>
<tr><td>Description</td><td colspan="5">Training 2D CNN federated model to classify prostate cancer images</td></tr>
<tr><td>POM</td><td>POM1</td><td>Algorithm</td><td>CNN</td><td>Quorum</td><td>2</td></tr>
<tr><td>Max Iterations</td><td>100</td><td>Pre-processing</td><td colspan="3">None in CC</td></tr>
<tr><th colspan="6"><u>Task execution</u></th></tr>
<tr><td rowspan="3" colspan="2">Aggregator</td><td>User</td><td colspan="3">b3d_node1</td></tr>
<tr><td>Val dataset</td><td colspan="3">uk_val_2D.pkl</td></tr>
<tr><td>Test dataset</td><td colspan="3">uk_tes_2D.pkl</td></tr>
<tr><td rowspan="6" colspan="2">Participants</td><td rowspan="4">User</td><td colspan="3">b3d_node2</td></tr>
<tr><td>Train dataset</td><td colspan="2">uk_tra_2D.pkl</td></tr>
<tr><td>Val dataset</td><td colspan="2">uk_val_2D.pkl</td></tr>
<tr><td>Test dataset</td><td colspan="2">uk_tes_2D.pkl</td></tr>
<tr><td rowspan="2">User</td><td colspan="3">b3d_node3</td></tr>
<tr><td>Train dataset</td><td colspan="2">gr_tra_2D.pkl</td></tr>
</table>

		Val dataset	gr_val_2D.pkl
		Test dataset	gr_tes_2D.pkl

The accuracy of the task is reported in the following table.

Task Results	
Accuracy b3d_node2 (Federated)	72.86%
Accuracy b3d_node3 (Federated)	80.93%

The confusion matrices for this task are presented in **Figure 13** and **Figure 14**.

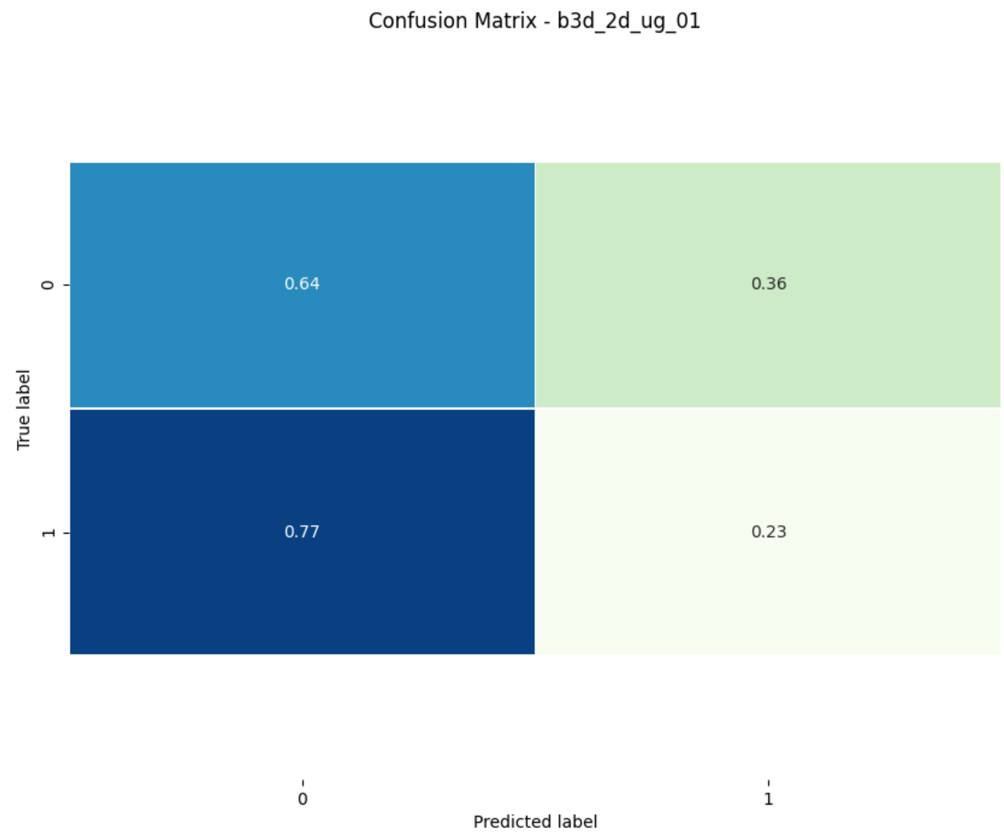

Figure 13. Confusion Matrix for 2D CNN with B3D + Hygeia data in Worker 1 B3D

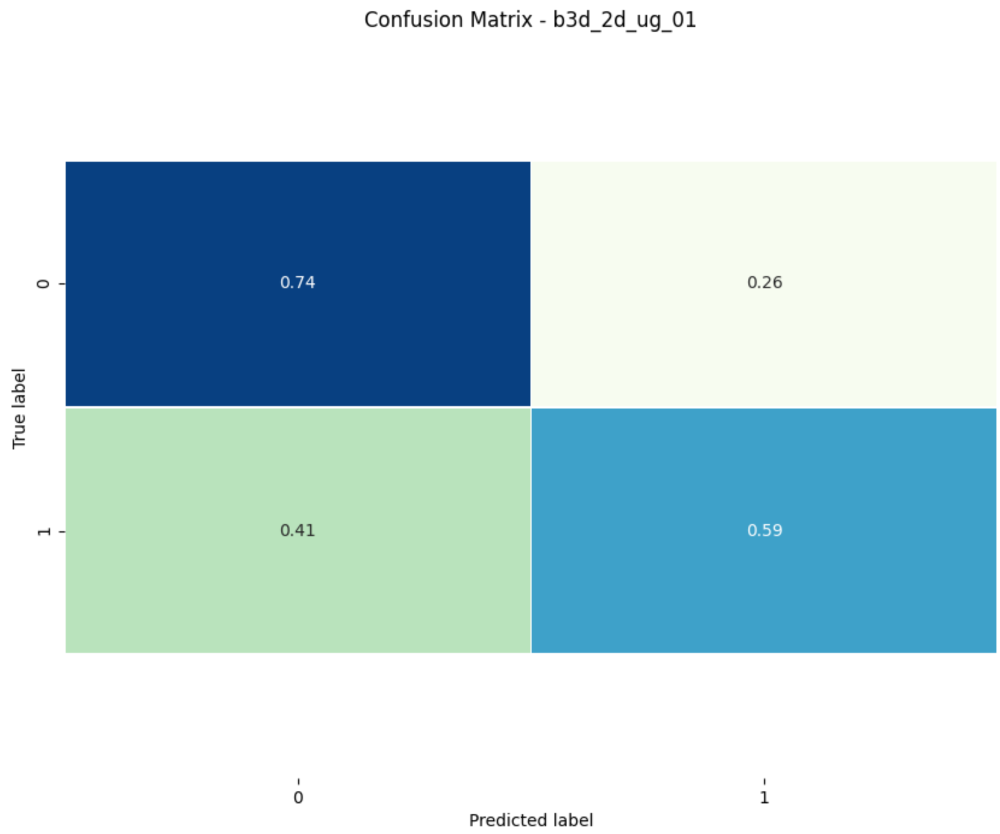

Figure 14. Confusion Matrix for 2D CNN with B3D + Hygeia data in Worker 2 Hygeia

4.4.2 Discussion of the 2D CNN model results

Table 6. Results of the 2D CNN tests

Model	Data	Worker	Accuracy	TP	TN	FP	FN
2D CNN	B3D	B3D	75.71%	0.48	0.64	0.36	0.52
2D CNN	Hygeia	Hygeia	70.43%	0.67	0.83	0.17	0.33
2D CNN	B3D+Hygeia	B3D	72.86%	0.23	0.64	0.36	0.77
2D CNN	B3D+Hygeia	Hygeia	80.93%	0.59	0.74	0.26	0.41

The accuracy of the model obtained with both datasets in Hygeia improved its accuracy above 10% in relation to the model trained only with Hygeia data. However, both True Positive and True Negative values decreased, lowering the sensitivity and specificity of the model.

The best 2D CNN predictive model in terms of sensitivity and specificity is the one trained only with Hygeia data:

- Sensitivity = (True Positive) / (True Positive + False Negative) = 67%

- Specificity = (True Negative) / (True Negative + False Positive) = 83%

4.4.3 3D CNN Results

TEST 1: B3D 3D Data

<table>
<tr><th colspan="6">Task definition</th></tr>
<tr><td>Task Name</td><td colspan="5">b3d_3D_u_02</td></tr>
<tr><td>Description</td><td colspan="5">Training 3D CNN federated model to classify prostate cancer imaging volumes</td></tr>
<tr><td>POM</td><td>POM1</td><td>Algorithm</td><td>CNN</td><td>Quorum</td><td>1</td></tr>
<tr><td>Max Iterations</td><td>100</td><td>Pre-processing</td><td colspan="3">None in CC</td></tr>
<tr><th colspan="6">Task execution</th></tr>
<tr><td rowspan="3">Aggregator</td><td>User</td><td colspan="4">b3d_node1</td></tr>
<tr><td>Val dataset</td><td colspan="4">uk_val_3D.pkl</td></tr>
<tr><td>Test dataset</td><td colspan="4">uk_tes_3D.pkl</td></tr>
<tr><td rowspan="4">Participants</td><td rowspan="4">User</td><td colspan="4">b3d_node2</td></tr>
<tr><td colspan="2">Train dataset</td><td colspan="2">uk_tra_3D.pkl</td></tr>
<tr><td colspan="2">Val dataset</td><td colspan="2">uk_val_3D.pkl</td></tr>
<tr><td colspan="2">Test dataset</td><td colspan="2">uk_tes_3D.pkl</td></tr>
</table>

The accuracy of the task is reported in the following table.

Task Results	
Accuracy b3d_node3 (Non-federated/Local)	75.56%

The confusion matrix for this task is presented in **Figure 15**.

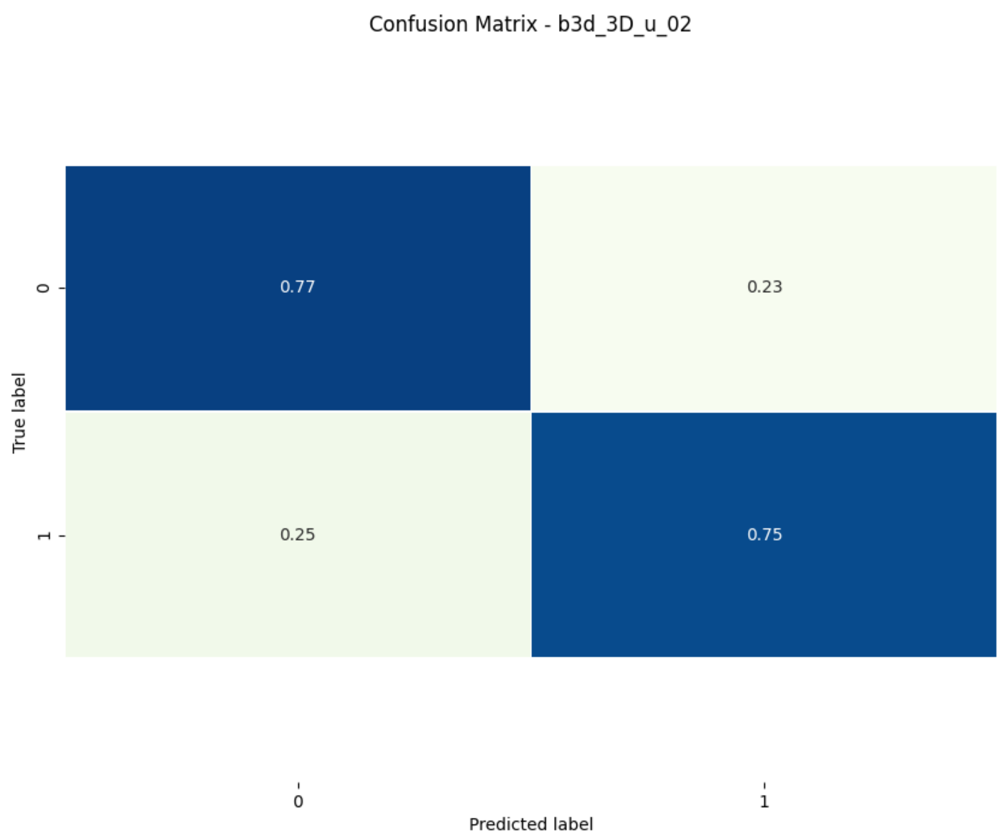

Figure 15. Confusion Matrix for 3D CNN with B3D data

TEST 2: Hygeia 3D data

<table>
<tr><th colspan="8"><u>Task definition</u></th></tr>
<tr><td>Task Name</td><td colspan="7">b3d_3D_g_05</td></tr>
<tr><td>Description</td><td colspan="7">Training 3D CNN federated model to classify prostate cancer imaging volumes</td></tr>
<tr><td>POM</td><td>POM1</td><td colspan="2">Algorithm</td><td colspan="2">CNN</td><td>Quorum</td><td>1</td></tr>
<tr><td>Max Iterations</td><td>100</td><td colspan="2">Pre-processing</td><td colspan="4">None in CC</td></tr>
<tr><th colspan="8"><u>Task execution</u></th></tr>
<tr><td colspan="2" rowspan="3">Aggregator</td><td>User</td><td colspan="5">b3d_node2</td></tr>
<tr><td>Val dataset</td><td colspan="5">gr_val_3D.pkl</td></tr>
<tr><td>Test dataset</td><td colspan="5">gr_tes_3D.pkl</td></tr>
<tr><td colspan="2" rowspan="4">Participants</td><td rowspan="4">User</td><td colspan="5">b3d_node3</td></tr>
<tr><td colspan="2">Train dataset</td><td colspan="3">gr_tra_3D.pkl</td></tr>
<tr><td colspan="2">Val dataset</td><td colspan="3">gr_val_3D.pkl</td></tr>
<tr><td colspan="2">Test dataset</td><td colspan="3">gr_tes_3D.pkl</td></tr>
</table>

The accuracy of the task is reported in the following table.

Task Results	
Accuracy b3d_node3 (Non-federated/Local)	70.00%

The confusion matrix for this task is presented in **Figure 16**.

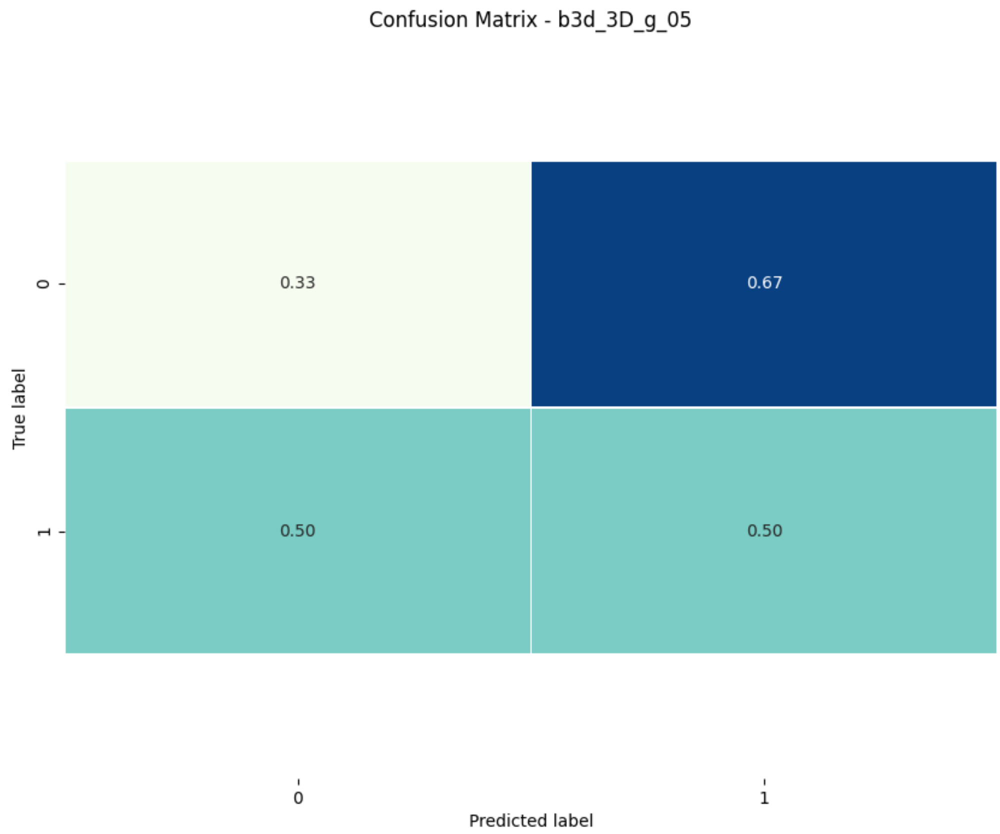

Figure 16. Confusion Matrix for 3D CNN with Hygeia data

TEST 3: B3D and Hygeia 3D data

Task definition					
Task Name	b3d_3D_ug_01				
Description	Training 3D CNN federated model to classify prostate cancer imaging volumes				
POM	POM1	**Algorithm**	CNN	**Quorum**	2
Max Iterations	100	**Pre-processing**	None in CC		
Task execution					
Aggregator	**User**	b3d_node1			
	Val dataset	uk_val_3D.pkl			
	Test dataset	uk_tes_3D.pkl			
Participants	**User**	b3d_node2			
		Train dataset	uk_tra_3D.pkl		

		Val dataset	uk_val_3D.pkl
		Test dataset	uk_tes_3D.pkl
	User	b3d_node3	
		Train dataset	gr_tra_3D.pkl
		Val dataset	gr_val_3D.pkl
		Test dataset	gr_tes_3D.pkl

The accuracy of the task is reported in the following table.

Task Results	
Accuracy b3d_node2 (Federated)	58.89%
Accuracy b3d_node3 (Federated)	67.00%

The confusion matrices for this task are presented in **Figure 17** and **Figure 18**.

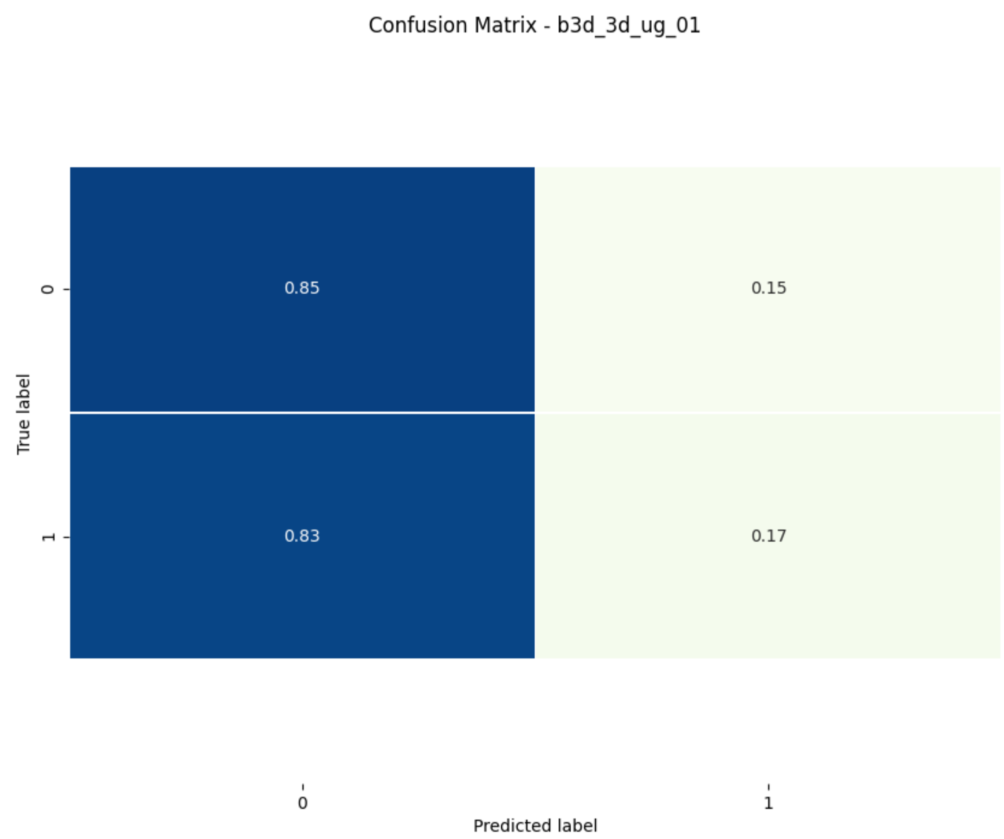

Figure 17. Confusion Matrix for 3D CNN with B3D + Hygeia data in Worker 1 B3D

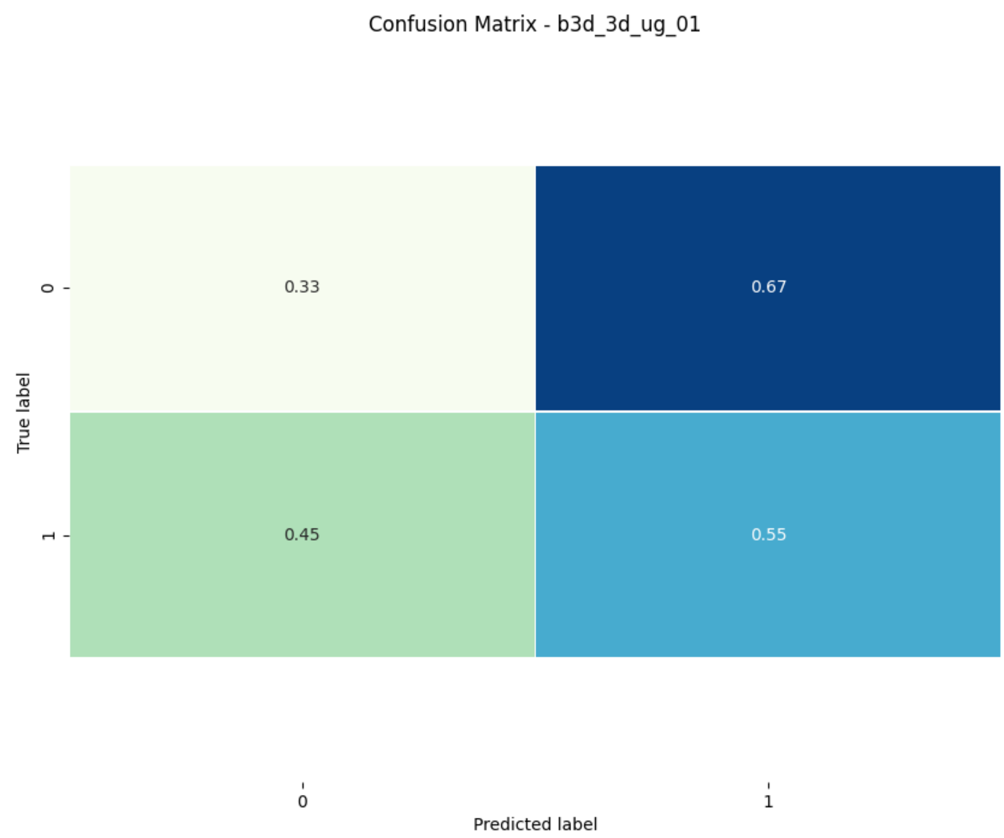

Figure 18. Confusion Matrix for 3D CNN with B3D + Hygeia data in Worker 2 Hygeia

4.4.4 Discussion of the 3D CNN model results

Table 7. Results of the 3D CNN tests

Model	Data	Worker	Accuracy	TP	TN	FP	FN
3D CNN	B3D	B3D	75.56%	0.75	0.77	0.23	0.25
3D CNN	Hygeia	Hygeia	70.00%	0.5	0.33	0.67	0.5
3D CNN	B3D+Hygeia	B3D	58.89%	0.17	0.85	0.15	0.83
3D CNN	B3D+Hygeia	Hygeia	67.00%	0.55	0.33	0.67	0.45

The accuracy of the model obtained with both datasets decreased its accuracy when comparing with the models trained only with B3D or Hygeia data. True Negative value increased for B3D model using both datasets, however, True Positive values decreased more significantly, lowering the sensitivity of the model. The sensitivity of the model trained with Hygeia data was improved by the use of both datasets.

The best 3D CNN predictive model in terms of sensitivity and specificity is the one trained only with B3D data:

- Sensitivity = (True Positive) / (True Positive + False Negative) = 75%
- Specificity = (True Negative) / (True Negative + False Positive) = 77%

From the point of view of the ML models the results are satisfactory at both sites (with slightly better results in Greece for the 2D model and better results in the UK for the 3D model). From the point of view of the platform it was also satisfactory because it was possible to obtain a combined trained model without the need to move data between the sites. While with the 2D models the FML improved the model accuracy, for the 3D models the accuracy decreased. In terms of sensitivity and specificity in both 2D and 3D models the FML decreased the predictive value.

5 Measurement Goals, Questions and Metrics

This section lists the goals, questions and metrics for the evaluation of the MUSKETEER results that are measured in the Health use case.

G1.2	
Analyse	MUSKETEER Architecture
For the purpose of	Evaluate
With respect to	standardization and extensibility
From the view point of	Business Perspective
In the context of	Health evaluation scenario (WP7)

The questions identified for the goal G1.2 are listed below.

Identifier	Questions
G1.2_Q01	Does it allow fast deployment and installation?
G1.2_Q02	Is it easy to use?
G1.2_Q03	Does it require special hardware locally?
G1.2_Q04	Does it allow interoperability with Medical Imaging Systems?

The goal G2.2 is described as follows:

G2.2	
Analyse	Privacy Preserving Operation Modes
For the purpose of	Evaluate

With respect to	privacy, computational overload, central storage requirements, communication requirements, data utility accountability
From the view point of	Business Perspective
In the context of	Health evaluation scenario (WP7)

The questions identified for the goal G2.2 are listed below.

Identifier	Questions
G2.2_Q01	How easy it is to verify and declare the privacy requirements?
G2.2_Q02	Due to our policy, is an adequate level of data privacy granted?
G2.2_Q03	Is it GDPR compliant?
G2.2_Q04	Is it compliant with Medical Devices' standards and regulations (MDR EU REGULATION 2017/745, EN ISO 13485:2016 and EN ISO 14971:2012)?
G2.2_Q05	Is it compliant with EN ISO/IEC 27001:2017?

The goal G3.3 is described as follows:

G3.3	
Analyse	Machine Learning Algorithms
For the purpose of	Evaluate
With respect to	pre-processing, normalization, data alignment, supervised and unsupervised learning
From the view point of	Business Perspective
In the context of	Health evaluation scenario (WP7)

The questions identified for the goal G3.3 are listed below.

Identifier	Questions
G3.3_Q01	Given the data on pelvis MRI exams as well as multiparametric MRI exams for male patients and a model adopted for predictions, is MUSKETEER able to improve the prediction model of the existence and grade of prostate cancer?
G3.3_Q02	Given the data on pelvis MRI exams as well as multiparametric MRI exams for male patients and a model adopted for predictions, is MUSKETEER trained model able to better identify and segment prostate cancer lesions?
G3.3_Q03	Data quality impacts on Machine Learning results. How is data quality controlled?

5.1 Metrics

Identifier	KPI	Format	Method of collection and measurement	Evaluation
G1.2_Q01_M01	(Time taken to deploy and install the MUSKETEER client) * (Number of employees involved to deploy and install the MUSKETEER client)	(HH:MM) * number of employees	Online questionnaire; face-to-face interview	24 HOURS * 2
G1.2_Q01_M02	(Time taken to update the MUSKETEER client) * (Number of employees involved update the MUSKETEER client)	(HH:MM) * number of employees	Online questionnaire; face-to-face interview	2 HOURS * 2
G1.2_Q02_M01	Time taken by one person to create a task	HH:MM	Online questionnaire; face-to-face interview	5 MINS
G1.2_Q02_M02	Time taken to run the training procedure associated to a given ML task	HH:MM	Online questionnaire; face-to-face interview	20 MINS
G1.2_Q02_M03	Time taken to select and use a trained ML model	HH:MM	Online questionnaire; face-to-face interview	20 MINS
G1.2_Q02_M04	Time required for training a new user	HH:MM	Online questionnaire; face-to-face interview	12 HOURS
G1.2_Q02_M05	Number of screens supported by help option / Total number of screens	Percentage	Verification of platform	81.25%
G1.2_Q03_M01	Cost of local special equipment	Number (EUR)	Online questionnaire; face-to-face interview	0 EUR
G1.2_Q03_M02	Cost of setting up local special equipment	Number (EUR)	Online questionnaire; face-to-face interview	0 EUR

G1.2_Q04_M01	Integration profile conformance statements	Number	Verification of platform/ technical documentation	2
G2.2_Q01_M01	Time taken by one person to verify and declare privacy requirements	HH:MM	Online questionnaire; fate-to-face interview	4 HOURS
G2.2_Q01_M02	Number of options expressed in natural language/ total number of steps	Decimal	Online questionnaire; fate-to-face interview	0.8
G2.2_Q02_M01	Adequate level of data privacy when sharing data with other end users	Boolean (true/false)	Online questionnaire; fate-to-face interview	TRUE (only model data is shared)
G2.2_Q02_M02	Adequate level of data privacy when sharing models with other end users	Boolean (true/false)	Online questionnaire; fate-to-face interview	TRUE
G2.2_Q02_M03	Adequate level of data privacy when training models	Boolean (true/false)	Online questionnaire; fate-to-face interview	TRUE
G2.2_Q03_M01	Implementation of GDPR requirements	Boolean (true/false)	Platform/ technical documentation verification	TRUE
G2.2_Q04_M01	Implementation of MDR, EN ISO 13485:2016 and EN ISO 14971:2012 security and privacy requirements	Boolean (true/false)	Platform/ technical documentation verification	TRUE
G2.2_Q04_M01	Implementation of EN ISO/IEC 27001:2017 requirements	Boolean (true/false)	Platform/ technical documentation verification	TRUE
G2.2_Q04_M02	Number of security controls implemented and verified	Number	Platform/ technical documentation verification	6
G3.3_Q01_M01	Sensitivity (true positive rate) of prediction of existence of cancer. Formula: Sensitivity = (True Positive) / (True Positive + False Negative)	percentage	Test ML models 2D CNN model 3D CNN model	67% 75%

G3.3_Q01_M02	Specificity (true negative rate) of prediction of existence of cancer. Formula: Specificity = (True Negative) / (True Negative + False Positive)	percentage	Test ML models 2D CNN model 3D CNN model	 83% 77%
G3.3_Q01_M03	Accuracy of classification of studies for each Patient	percentage	Test ML models 2D CNN model 3D CNN model	 80.93% 75.56%
G3.3_Q02_M01	Accuracy of Prostate segmentation	Percentage	Test ML models	N/A
G3.3_Q02_M02	Accuracy of lesion' segmentation	Percentage	Test ML models	N/A
G3.3_Q03_M01	Ratio of data errors in pre-processing, normalization, data alignment	Percentage	Tests in pre-processing, normalization, data alignment	5%
G3.3_Q03_M02	Ratio of empty data in pre-processing, normalization, data alignment	Percentage	Tests in pre-processing, normalization, data alignment	0%
G3.3_Q03_M03	Accuracy of ML trained model with dataset tested with external datasets	Percentage	Test ML models 2D CNN model 3D CNN model	 80.93% 67.00%

6 The Technical Perspective

G1.1	
Analyse	MUSKETEER Architecture
For the purpose of	Evaluate
With respect to	standardization and extensibility
From the view point of	Technical Perspective
In the context of	Use Cases validation (WP7)

The questions identified for the goal G1.1 are listed below.

Identifier	**Questions**

G1.1_Q01	Is the MUSKETEER architecture aligned with the Industrial Data Space Association reference architecture?
G1.1_Q02	Does it allow interoperability with ML frameworks?
G1.1_Q03	Does it foster the creation of a community of developers and researchers that can extend the platform with new algorithms and attack detection mechanisms?
G1.1_Q04	Does it allow fast deployment, installation and use?

G2.1	
Analyse	Privacy Preserving Operation Modes
For the purpose of	Evaluate
With respect to	privacy, computational overload, central storage requirements, communication requirements, data utility accountability
From the view point of	Technical Perspective
In the context of	Use Cases execution (WP7)

The questions identified for the goal G2.1 are listed below.

Identifier	Questions
G2.1_Q01	Will POMs be designed to allow a secure information exchange among platform user?
G2.1_Q02	Will POMs provide compliance with the legal and confidentiality restrictions of most industrial scenarios?
G2.1_Q03	Will the scalability of ML algorithms be improved over every POM, with regard to correct combination of different concepts of federated ML, differential privacy, homomorphic encryption, secure multiparty computation and distributed computing?

6.1 Metrics

Identifier	KPI	Format	Method of collection and measurement	Evaluation
G1.1_Q01_M01	Number of artifacts compliant with IDSA reference architecture / Total number of artifacts	Decimal <=1	Online questionnaire; fate-to-face interview (envisaged the involvement of IDSA in accordance with the DoW)	0.86
G1.1_Q02_M01	Number of ML libraries supported to	Decimal <=1	Online questionnaire;	1

	export the predictive models /Total of the best-known ML libraries		fate-to-face interview (the best-known ML libraries will be detailed more in the second version of this document)	
G1.1_Q03_M01	Open-source web communities' interactions	Integer	Field survey	4
G1.1_Q0.4_M01	Number of SW applications released as images	Integer	Field survey	4
G1.1_Q0.4_M02	Number of software components released in open-source repositories	Integer	Field survey	6
G2.1_Q01_M01	Number of robust POMs for use cases	Integer	On field	6
G2.1_Q02_M01	Speedup/number of users while POM is applied	Ratio	On field	POM depend-ent, see D6.2
G2.1_Q03_M01	Number of training procedures implemented	Integer	On field	33
G2.1_Q03_M02	Speed of privacy-preserving machine learning algorithms implemented with respect to other existing solutions	Ratio	On field	Not computable, see D6.2

7 Conclusion

Throughout the tasks T7.3 and T7.4 the partners involved in the Health use case have been able to deploy, test, validate and evaluate the MUSKETEER Federated Machine Learning platform, in a real-world scenario. This deliverable describes the health pilot for FML applied to medical imaging for supporting the diagnosis of prostate cancer. The deliverable presents preliminary steps of data selection, preparation and pre-processing, the selected ML algorithms and POMs, the setup of the MUSKETEER components, the training of the models and its results, and the evaluation of the MUSKETEER platform.

The partners involved in the Health Use Case selected relevant cases from the data available at their original sets. Data was anonymised and transformed into vectorised data, without any possible identifiable information. The partners choose to use Convolutional Neural Networks for classification of lesions in medical images from prostate cancer. The partners have used the MUSKETEER Machine Learning Library and the MUSKETEER Client Connector to create and train 2D and 3D models with the datasets available at Biotronics3D and Hygeia.

The 2D CNN models trained with both 2D datasets and both workers improved in accuracy when compared with 2D CNN models trained with each dataset separately. However, the 3D models trained with both 3D datasets decreased accuracy when compared with 3D CNN models trained with each B3D's dataset alone.

The partners involved in the Health use case, evaluated the platform based on the evaluation framework established in WP2. The evaluation of the business general performance indicators that enables to identify and measure the impact and the usefulness of the MUSKETEER platform was positive and met the criteria defined by the users. The Goal-Question-Metric approach enabled the Health use case partners to evaluate the MUSKETEER platform considering the architecture, the privacy preserving operation modes and the machine learning dimensions from a business perspective, with positive outcome. The technical partners performed the evaluation of the platform considering the architecture and the privacy preserving operation modes from the technical perspective, also following the Goal-Question-Metric approach, with positive results.

From the Health use case perspective, future work should explore the use of more data and more workers to increase the accuracy, the sensitivity and specificity of the models for use in the support to medical diagnosis. The security and privacy operation modes of the MUSKETEER platform enables us to persuade more organisations to participate in future experiments to increase the volume of annotated data and improve the performance of the models.

Annex A – 2D and 3D CNN models

2D CNN Model


```
{
    "class_name": "Sequential",
    "config": {
        "name": "sequential",
        "layers": [
            {
                "class_name": "InputLayer",
                "config": {
                    "batch_input_shape": [
                        null,
                        128,
                        128,
                        1
                    ],
                    "dtype": "float32",
                    "sparse": false,
                    "ragged": false,
                    "name": "conv2d_input"
                }
            },
            {
                "class_name": "Conv2D",
                "config": {
                    "name": "conv2d",
                    "trainable": true,
                    "batch_input_shape": [
                        null,
                        128,
                        128,
                        1
                    ],
                    "dtype": "float32",
                    "filters": 32,
                    "kernel_size": [
                        3,
                        3
                    ],
                    "strides": [
                        1,
                        1
                    ],
                    "padding": "valid",
                    "data_format": "channels_last",
                    "dilation_rate": [
                        1,
                        1
                    ],
                    "groups": 1,
                    "activation": "linear",
                    "use_bias": true,
                    "kernel_initializer": {
                        "class_name": "GlorotUniform",
```

```
                "config": {
                    "seed": null
                }
            },
            "bias_initializer": {
                "class_name": "Zeros",
                "config": {}
            },
            "kernel_regularizer": null,
            "bias_regularizer": null,
            "activity_regularizer": null,
            "kernel_constraint": null,
            "bias_constraint": null
        }
    },
    {
        "class_name": "Activation",
        "config": {
            "name": "activation",
            "trainable": true,
            "dtype": "float32",
            "activation": "relu"
        }
    },
    {
        "class_name": "MaxPooling2D",
        "config": {
            "name": "max_pooling2d",
            "trainable": true,
            "dtype": "float32",
            "pool_size": [
                2,
                2
            ],
            "padding": "valid",
            "strides": [
                2,
                2
            ],
            "data_format": "channels_last"
        }
    },
    {
        "class_name": "Conv2D",
        "config": {
            "name": "conv2d_1",
            "trainable": true,
            "dtype": "float32",
            "filters": 32,
            "kernel_size": [
                3,
                3
            ],
            "strides": [
                1,
```

```
                        1
                    ],
                    "padding": "valid",
                    "data_format": "channels_last",
                    "dilation_rate": [
                        1,
                        1
                    ],
                    "groups": 1,
                    "activation": "linear",
                    "use_bias": true,
                    "kernel_initializer": {
                        "class_name": "GlorotUniform",
                        "config": {
                            "seed": null
                        }
                    },
                    "bias_initializer": {
                        "class_name": "Zeros",
                        "config": {}
                    },
                    "kernel_regularizer": null,
                    "bias_regularizer": null,
                    "activity_regularizer": null,
                    "kernel_constraint": null,
                    "bias_constraint": null
                }
            },
            {
                "class_name": "Activation",
                "config": {
                    "name": "activation_1",
                    "trainable": true,
                    "dtype": "float32",
                    "activation": "relu"
                }
            },
            {
                "class_name": "MaxPooling2D",
                "config": {
                    "name": "max_pooling2d_1",
                    "trainable": true,
                    "dtype": "float32",
                    "pool_size": [
                        2,
                        2
                    ],
                    "padding": "valid",
                    "strides": [
                        2,
                        2
                    ],
                    "data_format": "channels_last"
                }
            },
```

```
{
	"class_name": "Conv2D",
	"config": {
		"name": "conv2d_2",
		"trainable": true,
		"dtype": "float32",
		"filters": 64,
		"kernel_size": [
			3,
			3
		],
		"strides": [
			1,
			1
		],
		"padding": "valid",
		"data_format": "channels_last",
		"dilation_rate": [
			1,
			1
		],
		"groups": 1,
		"activation": "linear",
		"use_bias": true,
		"kernel_initializer": {
			"class_name": "GlorotUniform",
			"config": {
				"seed": null
			}
		},
		"bias_initializer": {
			"class_name": "Zeros",
			"config": {}
		},
		"kernel_regularizer": null,
		"bias_regularizer": null,
		"activity_regularizer": null,
		"kernel_constraint": null,
		"bias_constraint": null
	}
},
{
	"class_name": "Activation",
	"config": {
		"name": "activation_2",
		"trainable": true,
		"dtype": "float32",
		"activation": "relu"
	}
},
{
	"class_name": "MaxPooling2D",
	"config": {
		"name": "max_pooling2d_2",
		"trainable": true,
```

```
                    "dtype": "float32",
                    "pool_size": [
                        2,
                        2
                    ],
                    "padding": "valid",
                    "strides": [
                        2,
                        2
                    ],
                    "data_format": "channels_last"
                }
            },
            {
                "class_name": "Flatten",
                "config": {
                    "name": "flatten",
                    "trainable": true,
                    "dtype": "float32",
                    "data_format": "channels_last"
                }
            },
            {
                "class_name": "Dense",
                "config": {
                    "name": "dense",
                    "trainable": true,
                    "dtype": "float32",
                    "units": 64,
                    "activation": "linear",
                    "use_bias": true,
                    "kernel_initializer": {
                        "class_name": "GlorotUniform",
                        "config": {
                            "seed": null
                        }
                    },
                    "bias_initializer": {
                        "class_name": "Zeros",
                        "config": {}
                    },
                    "kernel_regularizer": null,
                    "bias_regularizer": null,
                    "activity_regularizer": null,
                    "kernel_constraint": null,
                    "bias_constraint": null
                }
            },
            {
                "class_name": "Activation",
                "config": {
                    "name": "activation_3",
                    "trainable": true,
                    "dtype": "float32",
                    "activation": "relu"
```

```
                                }
                        },
                        {
                                "class_name": "Dropout",
                                "config": {
                                        "name": "dropout",
                                        "trainable": true,
                                        "dtype": "float32",
                                        "rate": 0.5,
                                        "noise_shape": null,
                                        "seed": null
                                }
                        },
                        {
                                "class_name": "Dense",
                                "config": {
                                        "name": "dense_1",
                                        "trainable": true,
                                        "dtype": "float32",
                                        "units": 1,
                                        "activation": "linear",
                                        "use_bias": true,
                                        "kernel_initializer": {
                                                "class_name": "GlorotUniform",
                                                "config": {
                                                        "seed": null
                                                }
                                        },
                                        "bias_initializer": {
                                                "class_name": "Zeros",
                                                "config": {}
                                        },
                                        "kernel_regularizer": null,
                                        "bias_regularizer": null,
                                        "activity_regularizer": null,
                                        "kernel_constraint": null,
                                        "bias_constraint": null
                                }
                        },
                        {
                                "class_name": "Activation",
                                "config": {
                                        "name": "activation_4",
                                        "trainable": true,
                                        "dtype": "float32",
                                        "activation": "sigmoid"
                                }
                        }
                ]
        },
        "keras_version": "2.4.0",
        "backend": "tensorflow"
}
```

3D CNN Model


```
{
	"class_name": "Sequential",
	"config": {
		"name": "sequential",
		"layers": [
			{
				"class_name": "InputLayer",
				"config": {
					"batch_input_shape": [
						null,
						16,
						64,
						64,
						1
					],
					"dtype": "float32",
					"sparse": false,
					"ragged": false,
					"name": "conv3d_input"
				}
			},
			{
				"class_name": "Conv3D",
				"config": {
					"name": "conv3d",
					"trainable": true,
					"batch_input_shape": [
						null,
						16,
						64,
						64,
						1
					],
					"dtype": "float32",
					"filters": 4,
					"kernel_size": [
						3,
						3,
						3
					],
					"strides": [
						1,
						1,
						1
					],
					"padding": "valid",
					"data_format": "channels_last",
					"dilation_rate": [
						1,
						1,
						1
					],
					"groups": 1,
```

```
                    "activation": "relu",
                    "use_bias": true,
                    "kernel_initializer": {
                        "class_name": "GlorotUniform",
                        "config": {
                            "seed": null
                        }
                    },
                    "bias_initializer": {
                        "class_name": "Zeros",
                        "config": {}
                    },
                    "kernel_regularizer": null,
                    "bias_regularizer": null,
                    "activity_regularizer": null,
                    "kernel_constraint": null,
                    "bias_constraint": null
                }
            },
            {
                "class_name": "MaxPooling3D",
                "config": {
                    "name": "max_pooling3d",
                    "trainable": true,
                    "dtype": "float32",
                    "pool_size": [
                        2,
                        2,
                        2
                    ],
                    "padding": "valid",
                    "strides": [
                        2,
                        2,
                        2
                    ],
                    "data_format": "channels_last"
                }
            },
            {
                "class_name": "Conv3D",
                "config": {
                    "name": "conv3d_1",
                    "trainable": true,
                    "batch_input_shape": [
                        null,
                        16,
                        64,
                        64,
                        1
                    ],
                    "dtype": "float32",
                    "filters": 8,
                    "kernel_size": [
                        3,
```

```
                        3,
                        3
                    ],
                    "strides": [
                        1,
                        1,
                        1
                    ],
                    "padding": "valid",
                    "data_format": "channels_last",
                    "dilation_rate": [
                        1,
                        1,
                        1
                    ],
                    "groups": 1,
                    "activation": "relu",
                    "use_bias": true,
                    "kernel_initializer": {
                        "class_name": "GlorotUniform",
                        "config": {
                            "seed": null
                        }
                    },
                    "bias_initializer": {
                        "class_name": "Zeros",
                        "config": {}
                    },
                    "kernel_regularizer": null,
                    "bias_regularizer": null,
                    "activity_regularizer": null,
                    "kernel_constraint": null,
                    "bias_constraint": null
                }
            },
            {
                "class_name": "MaxPooling3D",
                "config": {
                    "name": "max_pooling3d_1",
                    "trainable": true,
                    "dtype": "float32",
                    "pool_size": [
                        2,
                        2,
                        2
                    ],
                    "padding": "valid",
                    "strides": [
                        2,
                        2,
                        2
                    ],
                    "data_format": "channels_last"
                }
            },
```

```
                {
                        "class_name": "Flatten",
                        "config": {
                                "name": "flatten",
                                "trainable": true,
                                "dtype": "float32",
                                "data_format": "channels_last"
                        }
                },
                {
                        "class_name": "Dense",
                        "config": {
                                "name": "dense",
                                "trainable": true,
                                "dtype": "float32",
                                "units": 8,
                                "activation": "linear",
                                "use_bias": true,
                                "kernel_initializer": {
                                        "class_name": "GlorotUniform",
                                        "config": {
                                                "seed": null
                                        }
                                },
                                "bias_initializer": {
                                        "class_name": "Zeros",
                                        "config": {}
                                },
                                "kernel_regularizer": null,
                                "bias_regularizer": null,
                                "activity_regularizer": null,
                                "kernel_constraint": null,
                                "bias_constraint": null
                        }
                },
                {
                        "class_name": "Activation",
                        "config": {
                                "name": "activation",
                                "trainable": true,
                                "dtype": "float32",
                                "activation": "relu"
                        }
                },
                {
                        "class_name": "Dropout",
                        "config": {
                                "name": "dropout",
                                "trainable": true,
                                "dtype": "float32",
                                "rate": 0.5,
                                "noise_shape": null,
                                "seed": null
                        }
                },
```

```
                        {
                                "class_name": "Dense",
                                "config": {
                                        "name": "dense_1",
                                        "trainable": true,
                                        "dtype": "float32",
                                        "units": 1,
                                        "activation": "sigmoid",
                                        "use_bias": true,
                                        "kernel_initializer": {
                                                "class_name": "GlorotUniform",
                                                "config": {
                                                        "seed": null
                                                }
                                        },
                                        "bias_initializer": {
                                                "class_name": "Zeros",
                                                "config": {}
                                        },
                                        "kernel_regularizer": null,
                                        "bias_regularizer": null,
                                        "activity_regularizer": null,
                                        "kernel_constraint": null,
                                        "bias_constraint": null
                                }
                        }
                ]
        },
        "keras_version": "2.4.0",
        "backend": "tensorflow"
}
```