POLYTECH MONTPELLIER

UNIVERSITÉ DE MONTPELLIER

Mobility report S8

SDU

Syddansk Universitet

Student : Sarah THEOULLE
Internship tutor : Sadok BEN YAHIA
School tutor : Lysiane BUISSON LOPEZ

DO4
Year 2024-2025

Table of content

1. Introduction

As part of my computer engineering degree at Polytech Montpellier, I completed a three-month internship at the University of Southern Denmark in Sønderborg, Denmark, specifically in the Centre for Industrial Software (CIS).

The project I worked on involved the development of a specialized large language model (LLM) designed for multimodal data fusion to support breast cancer classification. This approach aims to improve diagnostic accuracy by leveraging the complementary strengths of different data modalities.

The main challenge I faced during this internship, apart from the difficulties of adapting to life in a foreign country, was learning to work with complex multimodal machine learning architectures, particularly managing the constraints of integrating diverse data types and adhering to the strict ethical and regulatory standards of medical research.

In terms of the structure of this report, I will first present the context of the internship and the challenges involved, describing the host organization, the work environment, the context and the request, as well as the required features. Next, I will describe the course of the internship, addressing the design and development, the project management methodology, the results obtained, and a critical analysis of my work. Finally, I will present an overview of the knowledge acquired and skills mobilized. To conclude, I will summarize the key points and future prospects.

2. Context of the Internship

2.1. Presentation of the host organization

I had the opportunity to do my internship at the University of Southern Denmark (SDU), one of the largest universities in Denmark. The University of Southern Denmark was established in 1998 with the fusion of multiple universities. SDU has six campuses, mainly located in the southern part of Denmark and the one where I work is the Alsion campus in Sønderborg, the southernmost campus in the country.

Image 1 : Sønderborg Alsion campus

The university offers a wide range of disciplines as well as a broad selection of business and engineering studies. The university focuses on areas such as communication, information technology, and biotechnology. Other areas of research are pursued through a number of national research centres at the university.

The Centre for Industrial Software (CIS) at the SDU Sønderborg campus is a software technological research and education centre, renowned for its dedication to innovation in computer science and digital technologies. My office within CIS specializes in artificial intelligence, with a particular focus on the development and application of large language models (LLMs) for a variety of research and industrial projects.

2.2. Context and request

During my internship, I worked under the supervision of my tutor, Professor Sadok Ben Yahia, and in close collaboration with a PhD student from Algeria. My role was to support her research by contributing to the development of a multimodal classification pipeline for breast cancer diagnosis.

The research context is at the intersection of artificial intelligence and medical imaging, where the aim is to exploit the complementarity of multiple data sources, such as mammography images, clinical metadata, and textual medical reports, to improve diagnostic performance. By combining several modalities within the same pipeline, the project seeks to enhance predictive power and provide more reliable decision support in breast cancer diagnosis.

2.3. Required features

The specific request made to me was twofold:

1. Develop a multimodal classification pipeline: This pipeline needed to integrate heterogeneous data types (images, structured clinical data, and text) in order to train deep learning models capable of classifying breast cancer cases. My task included data preprocessing, feature extraction from each modality, and implementing fusion strategies to combine the modalities.

2. Integrate the MCP protocol: This was an important part of the project. I needed to transform my work into an Agentic tool, using the MCP protocol. This is an open-source framework introduced in November 2024 to standardize the way artificial intelligence (AI) systems like large language models (LLMs) integrate and share data with external tools, systems, and data sources.

3. Course of the internship

3.1. Design and development

Despite the fact that I was working for a PhD student, I mainly worked alone on this project. When I arrived in Denmark, I only had basic knowledge of artificial intelligence. So I started by educating myself on the subject. The first few weeks of my internship were very enriching in this area. I was able to learn theoretically how LLM and different types of algorithms work, which proved useful later on.

The programming language I used throughout my entire internship was Python. It has a well-established ecosystem of high-quality libraries for data processing and matrix computation, which form the foundation of all artificial intelligence work.

Below is the technical architecture of my work throughout the first two months of my internship. The goal was to use a large number of datasets and train my models on them to grasp the technicalities of the project. For this I used 5 different datasets (in gray on the schema).

For the feature extraction and the training of my models I used various already trained LLMs that I finetuned for medical images classification. They are CNNs (convolutional neural networks). A convolutional neural network is a type of neural network that learns through recursive optimization. This type of deep learning network has been applied to process and make predictions from many different types of data including text, images and audio. This makes it very interesting for my work because of the large variety of data types we want to be able to inject into the model.

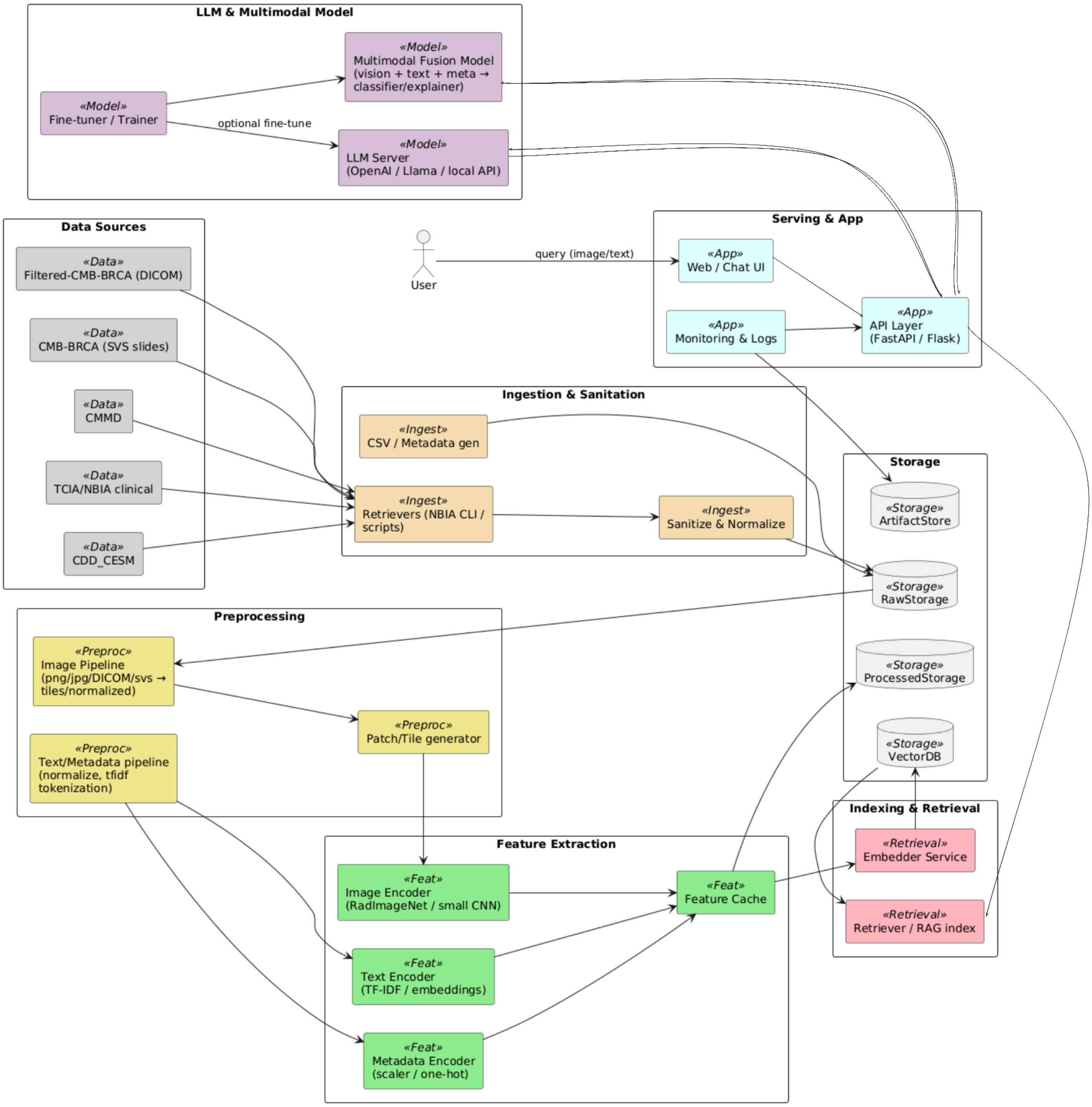


Figure 1 : technical architecture for the first months of the internship

The core of my work during this internship, which took place during my last month in Denmark, is implemented in the GitHub repository *Breast-classification-MCP* , which serves both as a research prototype and as a reproducible codebase for future extensions. The most important components are located in the `/agent` folder, which contains the implementation of the MCP Agent.

This agent has four main responsibilities:

1. **Data preprocessing**

    - Mammography or ultrasound images are normalized, resized, and augmented to improve generalization.

2. **Feature extraction**

    - A pretrained DenseNet-121 CNN (Densely Connected Convolutional Networks) is used for imaging features, leveraging transfer learning from large medical datasets.
    - Clinical data is encoded into structured numerical vectors.
    - Textual data is transformed into embeddings using language models.
    - Each branch can be trained independently.

3. **Multimodal fusion**

    - The extracted features are combined through concatenation strategies to form a unified representation.

4. **Evaluation and monitoring**

    - The pipeline computes classification metrics such as accuracy, F1-score, and ROC-AUC, and visualizes them through confusion matrices.
    - Results can be reproduced by rerunning the provided training scripts.

The repository is structured to encourage reproducibility:

- `requirements.txt` specifies the Python environment.
- `/agent` contains the modular MCP Agent code.
- Jupyter notebooks illustrate data exploration and preliminary experiments.

Finally, the integration of the MCP protocol transforms the multimodal classification pipeline into an *agentic tool*. This means the model can interact with external systems in a standardized way, making it possible to plug breast cancer classification into broader diagnostic or research workflows.

This implementation demonstrates not only the feasibility of multimodal fusion for breast cancer classification but also the importance of open and reproducible research practices.

Compared to a standalone pipeline, MCP offers significant advantages by providing a standardized interface that exposes tools such as `classify_breast_case` with clear JSON schemas. This means any MCP-aware client, whether an LLM, an application, or a script, can interact with the pipeline without requiring additional custom integration. It also improves reusability and governance: the same MCP tools can be used consistently across notebooks, backends, or chatbots, while requests and responses can be logged for auditability and reproducibility in research. Importantly, MCP enforces modularity, making it straightforward to introduce new modalities such as genomics or to swap in alternative models like vision transformers without altering the external contract.

3.2. Project management

3.2.1. Meetings

In order to ensure regular monitoring of the internship, I organized several meetings.

First, I met regularly with my mentor, Prof. Sadok, to review the progress of the project, discuss any obstacles encountered, and ask questions that were holding me back. These meetings allowed me to benefit from his experience and expert advice on the development side.

I also organized meetings with the person I was working for, Abir Belaala, to review my progress on the project, but above all to discuss together the data to be used and how to set up the project to best meet her needs.

3.2.2. Tools

To discuss with my team I used emails and teams for scheduled meetings.

The majority of my work has been done on my personal computer. However, when working on large datasets like I had to, the necessary resources were greater than those of my computer. Fortunately another department of SDU has a computer with a powerful GPU that I was able to share with other interns and PHDs.

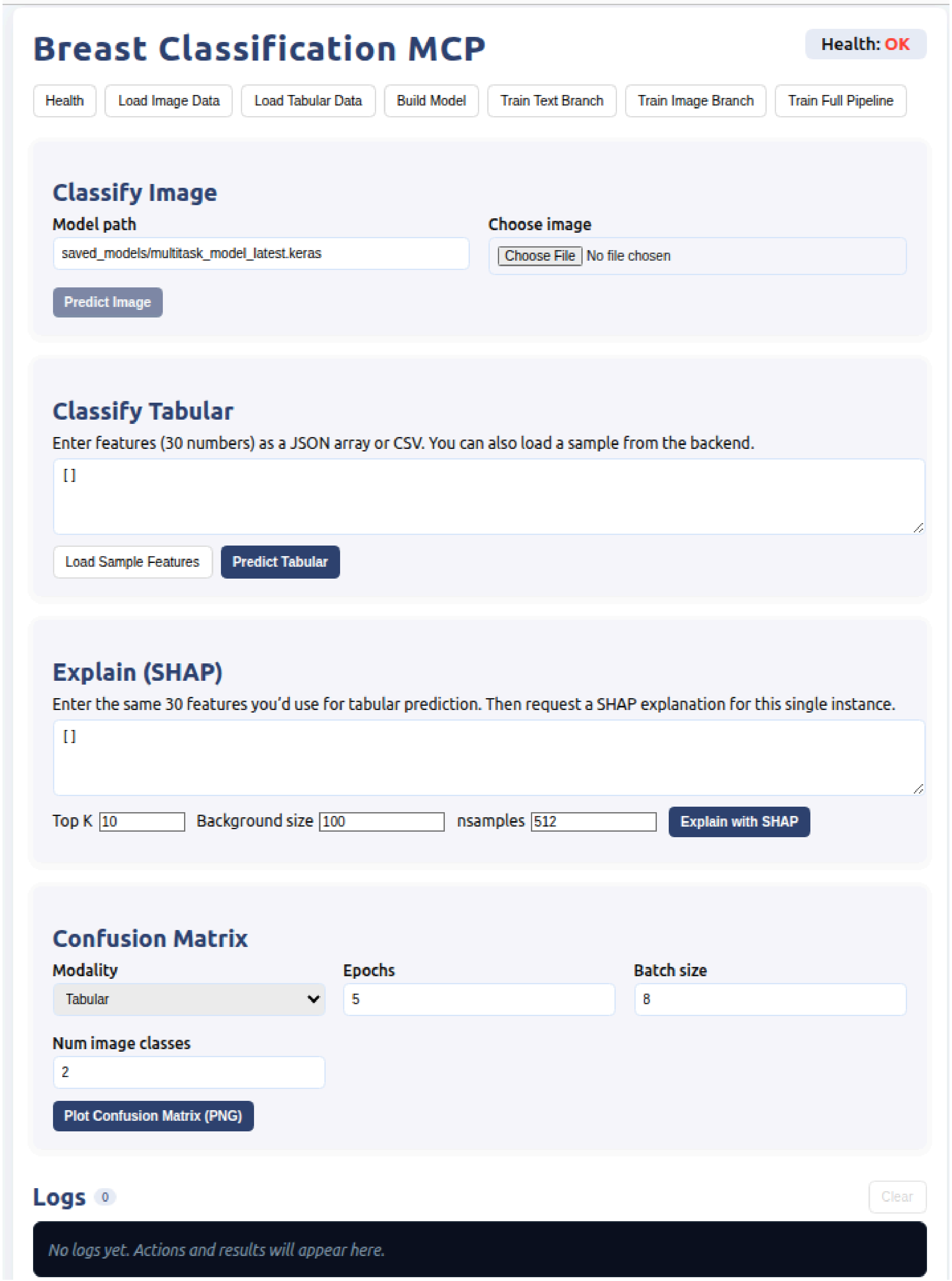
3.3. Results

In this part I will present the results I obtained for the last part of my internship. All results from the previous months are available at project_LLM but are not relevant to the product I delivered other than as evidence of my learning process and initial experimentations.

The image branch of the pipeline is trained on a three-class classification problem, distinguishing between benign, malignant, and normal cases. The dataset for this branch contains 1,262 training samples and 316 validation samples, with the classes mapped as follows: benign is class 0, malignant is class 1, and normal is class 2.

In contrast, the tabular branch handles a binary classification task, focusing only on benign versus malignant cases. Its dataset is smaller, with 455 training samples and 114 validation samples, and the class mapping assigns benign as class 0 and malignant as class 1.

Figure 1 shows the final frontend interface I developed. This interface allows users to interact with the model, visualize predictions, and access interpretability features such as SHAP explanations in real time. The frontend was a key component in delivering a usable product, connecting the AI model to users.



Breast Classification MCP

Health: OK

Classify Image

Model path: saved_models/multitask_model_latest.keras

Choose image: No file chosen

Classify Tabular

Enter features (30 numbers) as a JSON array or CSV. You can also load a sample from the backend.

```
[]
```

Explain (SHAP)

Enter the same 30 features you'd use for tabular prediction. Then request a SHAP explanation for this single instance.

```
[]
```

Top K: 10 Background size: 100 nsamples: 512

Confusion Matrix

Modality: Tabular

Epochs: 5

Batch size: 8

Num image classes: 2

Logs 0

No logs yet. Actions and results will appear here.

Figure 2 : Frontend in React

Tabular branch

The tabular model achieved excellent results, as summarized in Table 1. The confusion matrix (Figure 2) highlights that the model rarely confused benign and malignant classes, leading to an overall accuracy of **97%**. Both classes are well balanced, with macro-averaged precision, recall, and F1-scores of 0.97.

These results demonstrate the strong discriminative ability of the tabular model when applied to structured input features.

Class	Precision	Recall	F1-score	Support
0 (Benign)	0.95	0.98	0.97	43
1 (Malignant)	0.99	0.97	0.98	71
Accuracy			**0.97**	114
Macro avg	0.97	0.97	0.97	114
Weighted avg	0.97	0.97	0.97	114

Table 1 : Results of tabular model

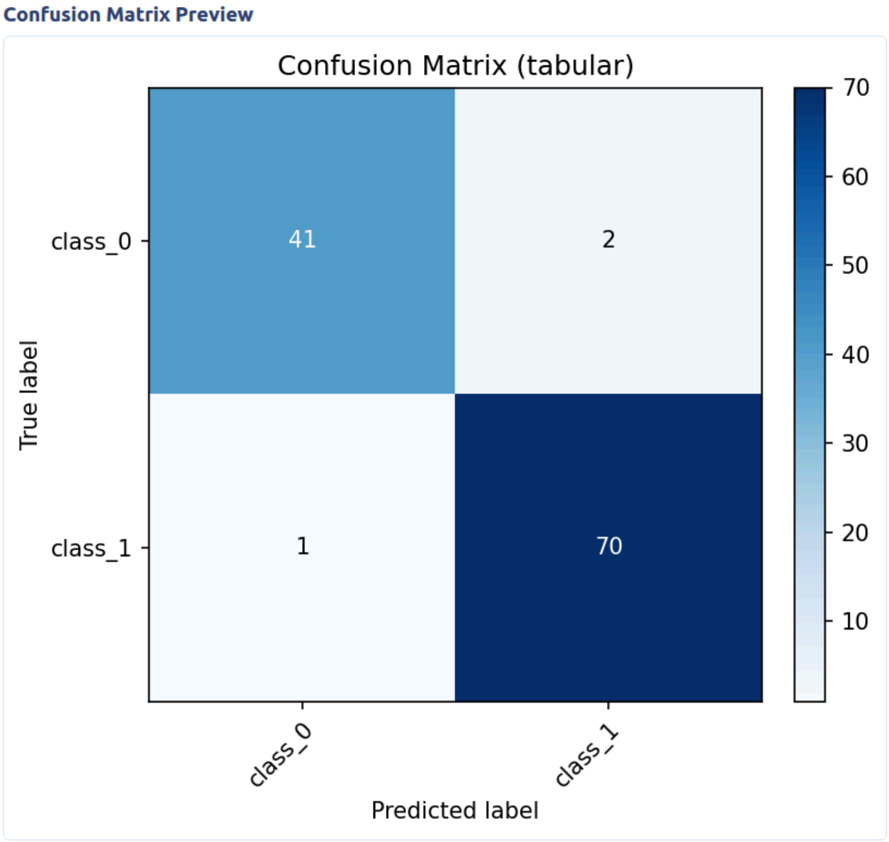


Figure 3 : Confusion matrix for tabular data

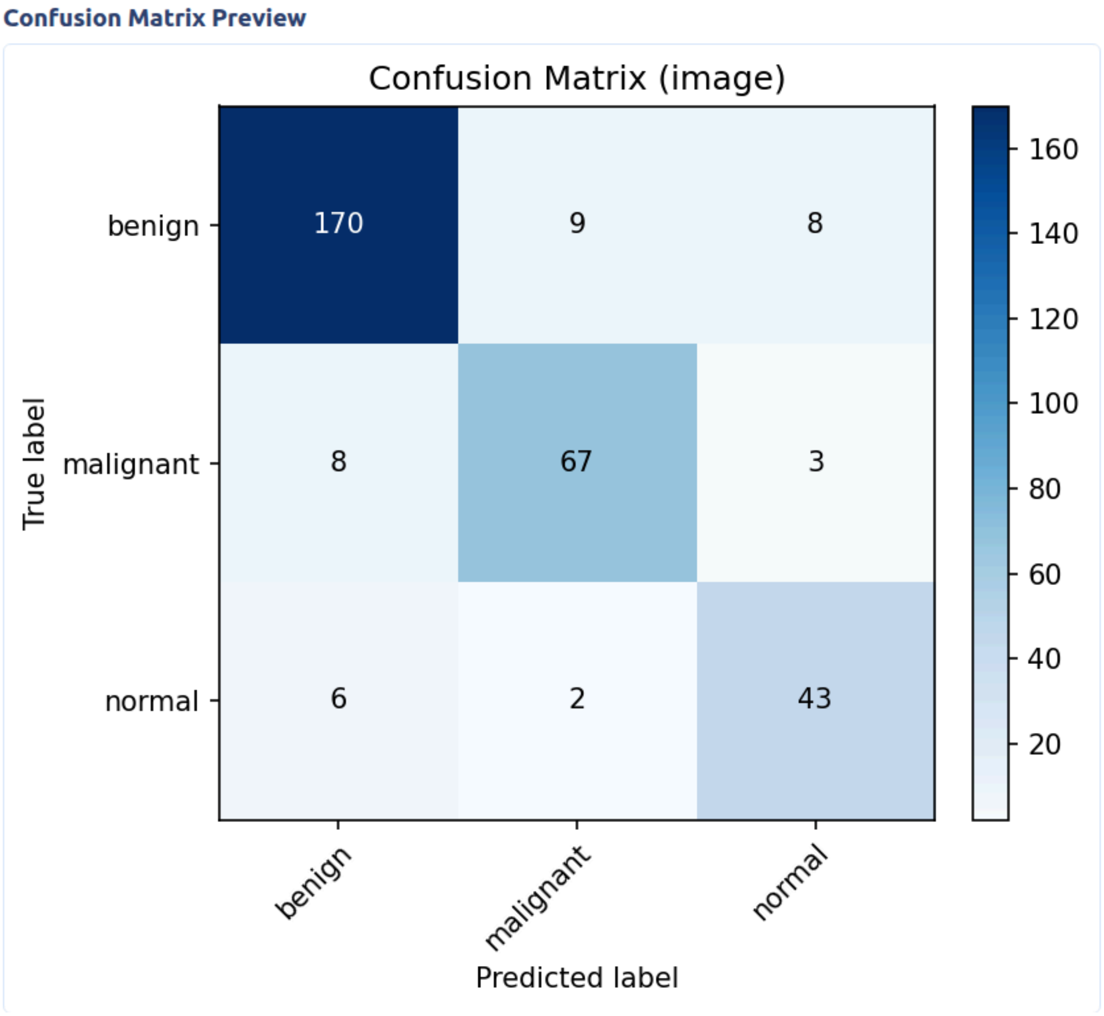
Image branch

In contrast, the image branch obtained lower performance, with an overall accuracy of **89%** (Table 2). The confusion matrix (Figure 3) indicates that benign lesions were classified reliably (recall = 0.96), while the model struggled more with malignant vs. normal separation.

These results are consistent with the expected challenges of medical image classification on limited datasets: malignant and normal tissue often share subtle visual patterns that make them harder to distinguish.

Class	Precision	Recall	F1-score	Support
Benign	0.89	0.96	0.92	187
Malignant	0.93	0.82	0.87	78
Normal	0.84	0.73	0.78	51
Accuracy			**0.89**	316
Macro avg	0.89	0.84	0.86	316
Weighted avg	0.89	0.89	0.89	316

Table 2 : Results of images data

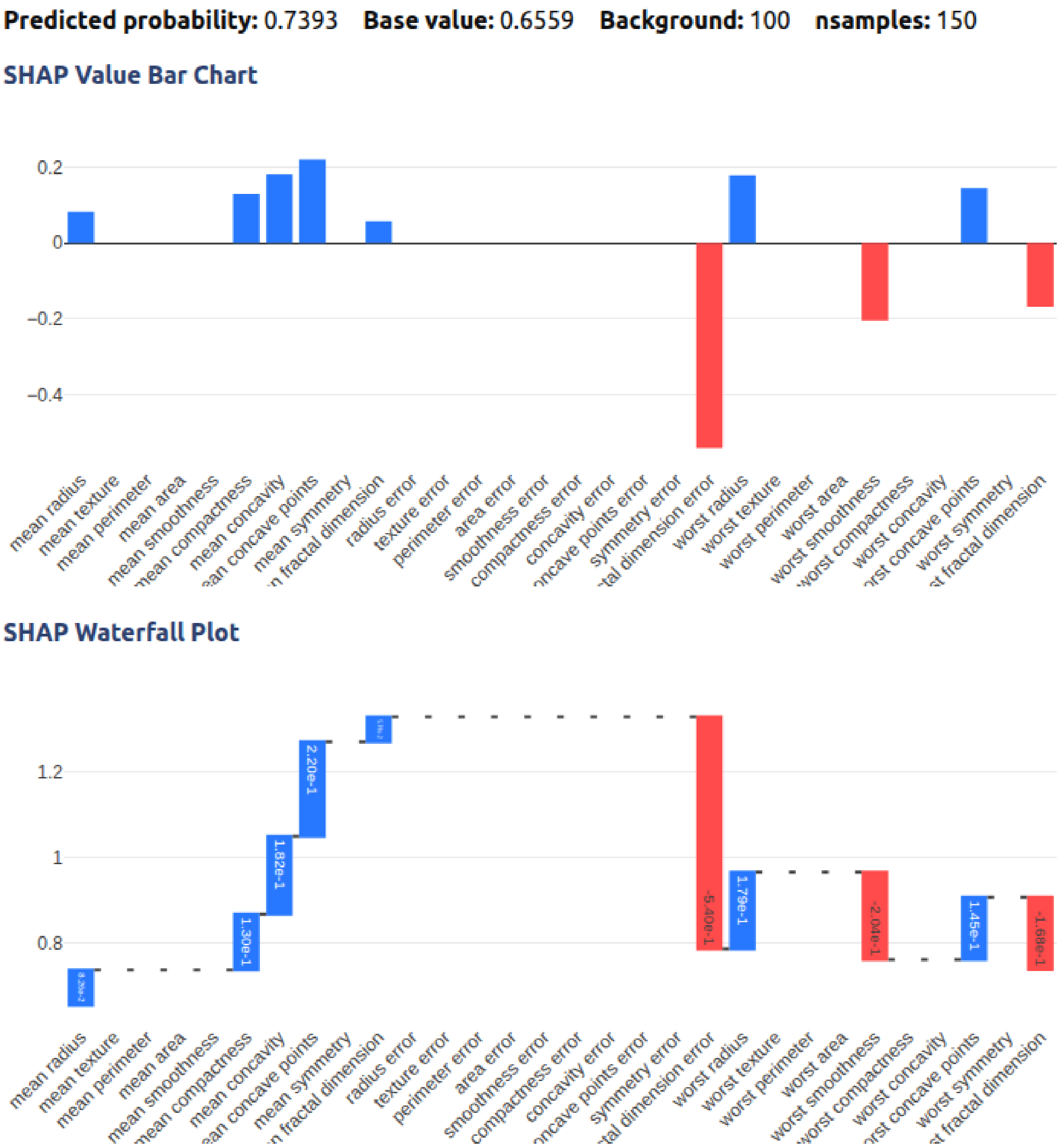


True label \ Predicted label	benign	malignant	normal
benign	170	9	8
malignant	8	67	3
normal	6	2	43

Figure 4 : Confusion matrix of images data

To improve the transparency of the system, SHAP (SHapley Additive exPlanations) was used to analyze feature contributions.

- **Per-sample explanation (Figure 4):** The SHAP waterfall plot illustrates how individual features influenced the model's decision for a given patient. Red bars correspond to features increasing the likelihood of malignancy, while blue bars indicate features lowering this probability (toward benign). This provides patient-specific interpretability.



Predicted probability: 0.7393 **Base value:** 0.6559 **Background:** 100 **nsamples:** 150

SHAP Value Bar Chart

SHAP Waterfall Plot

Figure 5 : Per-sample SHAP plots

- **Global explanation (Figure 5):** The SHAP dependence plots summarize feature impact across the dataset. Each point corresponds to a patient sample, showing how increasing values of certain clinical features consistently push predictions toward malignant. This aligns with clinical expectations and validates the model's decision-making process.

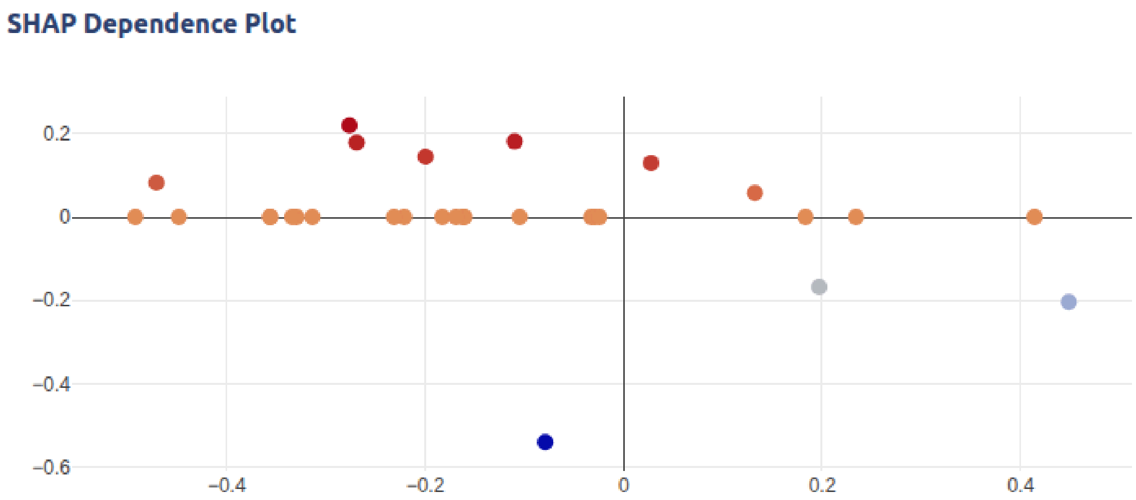


Figure 6 : SHAP dependence plot

DenseNet-121 demonstrated strong performance in classifying breast ultrasound images, though with lower accuracy compared to the structured-data branch. The use of transfer learning and data augmentation proved essential in handling the limited dataset size and variability.

State of the art comparison

Several studies have explored deep learning approaches for breast ultrasound image classification, with DenseNet-based architectures. For instance, Omkarmodi et al. [1] reported a test accuracy of 86% on a dataset of 100 samples, achieving very high recall for benign lesions (0.98) but substantially lower performance for malignant cases (recall = 0.67). Aliabedimadiseh et al. [2] and Rentsi et al. [3] both evaluated DenseNet121 on 117 test images, obtaining comparable accuracies of 87%. Their models demonstrated excellent detection of benign and normal tissue (recall up to 1.00), but consistently struggled to identify malignant lesions, with recalls limited to 0.65.

In contrast, my image branch was evaluated on a larger and more diverse test set of 316 cases (I split the dataset between train and test instead of training, evaluation and test). It achieved an overall accuracy of 89% and a macro F1-score of 0.86, which is on par or superior to previously reported DenseNet baselines. Importantly, my model reached a malignant recall of 0.82, substantially higher than the 0.65–0.67 range observed in prior work. This improvement is clinically significant, as malignant lesion detection is the most critical aspect of breast cancer diagnosis. While my method showed somewhat lower recall for the normal class (0.73 versus up to 1.00 in other reports), it achieved a more balanced trade-off across all classes, suggesting stronger generalizability when applied to larger cohorts.

3.4. Critical analysis

Throughout the internship, I faced several challenges:

- Technical challenges:
  - Handling heterogeneous data modalities (images, text, metadata) required building preprocessing pipelines that could normalize and align different feature spaces.
  - The limited size and imbalance of medical datasets led to risks of overfitting, making it necessary to use transfer learning, data augmentation, and careful validation strategies.
  - Integration of the MCP protocol into the pipeline: it required both a solid understanding of the AI agent architecture and compliance with a newly emerging standard.

- Computational challenges:
  - Training deep models on medical images demanded GPU resources that exceeded my personal computer's capacity. I had to adapt my workflow to use shared high-performance computing resources at SDU.

- Organizational challenges:
  - Working mostly independently, I needed to develop strong self-management skills to structure my progress and ensure regular reporting to my supervisor.

The project is situated in the field of medical AI, where both sustainability and social responsibility are key concerns:

- Sustainable development:

  - Medical AI requires significant computational resources, which can have an important carbon footprint. During my internship, I navigated this by using transfer learning and already existing models, instead of training models from scratch, reducing training time and energy consumption.
  - I adopted resource-sharing practices, relying on shared GPU infrastructure rather than duplicating hardware usage.

- Social responsibility:

  - Breast cancer is a major global health issue, and improving diagnostic tools directly contributes to United Nations Sustainable Development Goal 3 (Good Health and Well-Being).
  - The integration of multimodal data aims to improve diagnostic accuracy, thus reducing unnecessary biopsies and improving patient outcomes.
  - Ethical considerations were always present: respecting data privacy regulations (GDPR) by using anonymized data, avoiding algorithmic bias, and

ensuring that the tool remains an assistive technology for clinicians rather than a replacement.

Overall, this project illustrates how responsible AI practices can combine technical innovation with social impact, ensuring that progress in machine learning serves both patients and the medical community in an ethical and sustainable way.

4. Conclusion

My internship at the University of Southern Denmark offered me the opportunity to explore a really interesting research area, at the intersection of artificial intelligence and medical imaging. Over the course of three months, I contributed to the design and implementation of a multimodal classification pipeline for breast cancer diagnosis, integrating heterogeneous data sources such as medical images, clinical metadata, and textual reports.

From a technical perspective, I gained knowledge of machine learning and artificial intelligence, acquiring practical skills in data preprocessing, model training, multimodal fusion, and evaluation. I also gained hands-on experience with advanced tools such as convolutional neural networks, transfer learning, and interpretability techniques like SHAP. Working with the MCP protocol encouraged me to think past traditional standalone models and focus on building adaptable, reusable solutions that respect evolving standards.

On a personal level, this internship strengthened my autonomy, adaptability, and ability to collaborate in an international research environment. Working largely independently while benefiting from regular guidance taught me to structure my work and communicate effectively with both my supervisor and collaborators.

The results obtained demonstrate both the potential and the limitations of multimodal approaches in breast cancer classification. While the tabular branch reached high accuracy and interpretability, the image branch highlighted the challenges of working with limited medical datasets.

Overall, this internship has been an essential step in my academic and professional journey, consolidating my technical skills, enriching my cultural and personal experience, and confirming my motivation to pursue research and innovation in artificial intelligence applied to healthcare.

5. Acknowledgments

I would like to express my deepest gratitude to everyone who contributed in any way to the success of my internship.

First, I would like to thank Polytech Montpellier and the entire teaching team, whose guidance and support enabled me to be here today.

In particular, I would like to thank Professor Sadok Ben Yahia for his help, expertise, teaching skills, and kindness throughout the internship. My work would not have reached its current quality without his invaluable advice.

I am also grateful to Benoît, Etienne, Corentin, Nassim, Tarek, and Kyle, my colleagues with whom I carried out part or all of this internship, for their warm welcome, support, and collaboration throughout this work.

Finally, I would like to thank my family and friends for their constant encouragement during this period and for proofreading this report.

6. List of figures and bibliography

[1] Omkarmodi, *DenseNet for Breast Ultrasound Classification*, Kaggle Notebook, 2023.
[2] Aliabedimadiseh, *DenseNet121 for Breast Ultrasound Classification*, Kaggle Notebook, 2023.
[3] Rentsi, *Breast Cancer Ultrasound Classification with DenseNet*, Kaggle Notebook, 2023.

References and Resources used

[1] Mingzhe Hu, Joshua Qian, Shaoyan Pan, Yuheng Li, Richard Qiu, Xiaofeng Yang. "Advancing medical imaging with language models: featuring a spotlight on ChatGPT." Physics in Medicine & Biology, 2024. https://doi.org/10.1088/1361-6560/ad387d

[2] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, Eftychios Protopapadakis. "Deep Learning for Computer Vision: A Brief Review." Computational Intelligence and Neuroscience, 2018. https://onlinelibrary.wiley.com/doi/abs/10.1155/2018/7068349

[3] Abdullah Ayub Khan, Asif Ali Laghari, Shafique Ahmed Awan. “Machine Learning in Computer Vision: A Review.” EAI Endorsed Transactions on Scalable Information Systems, 2021. https://publications.eai.eu/index.php/sis/article/view/2055

[4] Model Context Protocol. Wikipédia, 2025. https://fr.wikipedia.org/w/index.php?title=Model_Context_Protocol&oldid=226544430

[5] Retrieval-augmented generation. Wikipedia, 2025. https://en.wikipedia.org/w/index.php?title=Retrieval-augmented_generation&oldid=1296772806

[6] Rhiannon Williamsarchive. “Why Google’s AI Overviews gets things wrong.” MIT Technology Review, 2024. https://www.technologyreview.com/2024/05/31/1093019/why-are-googles-ai-overviews-results-so-bad/

[7] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.” 2020.

[8] Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. 2020. https://dl.acm.org/doi/pdf/10.5555/3495724.3496517

[9] Remote MCP - OpenAI API. https://platform.openai.com

[10] Jason Weston, Sumit Chopra, Antoine Bordes. “Memory Networks.” arXiv, 2015. http://arxiv.org/abs/1410.3916

[11] Jason Weston, Sumit Chopra & Antoine Bordes, MEMORY NETWORK, 2015 https://arxiv.org/pdf/1410.3916

[12] Sainbayar Sukhbaatar. “End-To-End Memory Networks.” 2015.

[13] Kenton Lee, Ming-Wei Chang, Kristina Toutanova. “Latent Retrieval for Weakly Supervised Open Domain Question Answering.” Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019. https://www.aclweb.org/anthology/P19-1612

[14] Latent Retrieval for Weakly Supervised Open Domain Question Answering, 2019 https://aclanthology.org/P19-1612.pdf

[15] Unsupervised learning. Wikipedia, 2025. https://en.wikipedia.org/w/index.php?title=Unsupervised_learning&oldid=1288079332

[16] Supervised learning. Wikipedia, 2025. https://en.wikipedia.org/w/index.php?title=Supervised_learning&oldid=1282767374

[17] Jeffrey Ip. “The Definitive LLM-as-a-Judge Guide for Scalable LLM Evaluation.” Medium, 2025.

https://medium.com/@jeffreyip54/the-definitive-llm-as-a-judge-guide-for-scalable-llm-evaluation-a4aad7b455b9

[18] Sergiusz Łukasiewicz, Marcin Czeczelewski, Alicja Forma, Jacek Baj, Robert Sitarz, Andrzej Stanisławek. “Breast Cancer—Epidemiology, Risk Factors, Classification, Prognostic Markers, and Current Treatment Strategies—An Updated Review.” Cancers, 2021. https://www.mdpi.com/2072-6694/13/17/4287

[19] What Is ResNet-50? Roboflow Blog, 2024. https://blog.roboflow.com/what-is-resnet-50/

[20] How to Train and Deploy a ResNet-50 Model. Roboflow Blog, 2025. https://blog.roboflow.com/how-to-train-a-resnet-50-model/

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. “Deep Residual Learning for Image Recognition.” arXiv, 2015. http://arxiv.org/abs/1512.03385

[22] imagePretrainedNetwork - Pretrained neural network for images - MATLAB. https://fr.mathworks.com/help/deeplearning/ref/imagepretrainednetwork.html

[23] Models and pre-trained weights — Torchvision main documentation. https://docs.pytorch.org/vision/main/models.html

[24] Vera Sorin, Benjamin S. Glicksberg, Yaara Artsi, Yiftach Barash, Eli Konen, Girish N. Nadkarni, Eyal Klang. “Utilizing large language models in breast cancer management: systematic review.” Journal of Cancer Research and Clinical Oncology, 2024. https://link.springer.com/10.1007/s00432-024-05678-6

[25] Appropriateness of Breast Cancer Prevention and Screening Recommendations Provided by ChatGPT. https://pubs.rsna.org/doi/epdf/10.1148/radiol.230424

[26] Sebastian Griewing, Niklas Gremke, Uwe Wagner, Michael Lingenfelder, Sebastian Kuhn, Jelena Boekhoff. “Challenging ChatGPT 3.5 in Senology—An Assessment of Concordance with Breast Cancer Tumor Board Decision Making.” Journal of Personalized Medicine, 2023. https://www.mdpi.com/2075-4426/13/10/1502

[27] Arya Rao, John Kim, Meghana Kamineni, Michael Pang, Winston Lie, Keith J. Dreyer, Marc D. Succi. “Evaluating GPT as an Adjunct for Radiologic Decision Making: GPT-4 Versus GPT-3.5 in a Breast Imaging Pilot.” Journal of the American College of Radiology, 2023. https://www.jacr.org/article/S1546-1440(23)00394-0/abstract

[28] Hyeon Seok Choi, Jun Yeong Song, Kyung Hwan Shin, Ji Hyun Chang, Bum-Sup Jang. “Developing prompts from large language model for extracting clinical information from pathology and ultrasound reports in breast cancer.” Radiation Oncology Journal, 2023. https://doi.org/10.3857/roj.2023.00633

[29] Stefan Lukac, Davut Dayan, Visnja Fink, Elena Leinert, Andreas Hartkopf, Kristina Veselinovic, Wolfgang Janni, Brigitte Rack, Kerstin Pfister, Benedikt Heitmeir, Florian Ebner. “Evaluating ChatGPT as an adjunct for the multidisciplinary tumor board decision-making in

primary breast cancer cases." Archives of Gynecology and Obstetrics, 2023. https://doi.org/10.1007/s00404-023-07130-5

[30] A Large Language Model Pipeline for Breast Cancer Oncology. https://arxiv.org/html/2406.06455v2

[31] Binomial proportion confidence interval. Wikipedia, 2025. https://en.wikipedia.org/w/index.php?title=Binomial_proportion_confidence_interval&oldid=1291256679

[32] Raphael Pelossof, Mark Carty, Talal Ahmed, Stanislas Lauly, Alberto Purpura, Erik Mueller, Justin Guinney. "Abstract 5006: Multi-modal large language models for metastatic breast cancer prognosis." Cancer Research, 2025. https://doi.org/10.1158/1538-7445.AM2025-5006

[33] Loic Ah-thiane, Pierre-Etienne Heudel, Mario Campone, Marie Robert, Victoire Brillaud-Meflah, Caroline Rousseau, Magali Le Blanc-Onfroy, Florine Tomaszewski, Stéphane Supiot, Tanguy Perennec, Augustin Mervoyer, Jean-Sébastien Frenel. "Large Language Models as Decision-Making Tools in Oncology: Comparing Artificial Intelligence Suggestions and Expert Recommendations." JCO Clinical Cancer Informatics, 2025. https://ascopubs.org/doi/10.1200/CCI-24-00230

[34] Complete Guide to Building a Transformer Model with PyTorch. https://www.datacamp.com/tutorial/building-a-transformer-with-py-torch

[35] PacktPublishing/Mastering-Transformers. Packt, 2025. https://github.com/PacktPublishing/Mastering-Transformers

[36] Réseau de neurones récurrents. Wikipédia, 2025. https://fr.wikipedia.org/w/index.php?title=R%C3%A9seau_de_neurones_r%C3%A9currents&oldid=225052042

[37] Algorithme du gradient. Wikipédia, 2025. https://fr.wikipedia.org/w/index.php?title=Algorithme_du_gradient&oldid=225899873

[38] Rétropropagation du gradient. Wikipédia, 2025. https://fr.wikipedia.org/w/index.php?title=R%C3%A9tropropagation_du_gradient&oldid=226469347

[39] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Perric Cistac, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, Alexander M. Rush. "Transformers: State-of-the-Art Natural Language Processing." Association for Computational Linguistics, 2020. https://www.aclweb.org/anthology/2020.emnlp-demos.6

[40] Model Context Protocol (MCP) in Pharma. IntuitionLabs, 2025. https://intuitionlabs.ai/articles/model-context-protocol-mcp-in-pharma

[41] Best Practices for Randomization and Trial Supply Management (RTSM) in Phase 3 Clinical Trials. IntuitionLabs, 2025. https://intuitionlabs.ai/articles/best-practices-for-randomization-and-trial-supply-management-rtsm-in-phase-3-clinical-trials

[42] vishrut-b/ML-Project-with-PyTorch-Breast-Cancer-Classification. https://github.com/vishrut-b/ML-Project-with-PyTorch-Breast-Cancer-Classification/tree/main

[43] Sarah THEOULLE. “stheoulle/Breast-Cancer-Image-Classification-with-DenseNet121.” GitHub, 2025. https://github.com/stheoulle/Breast-Cancer-Image-Classification-with-DenseNet121

[44] Model Context Protocol (MCP) Tutorial: Build Your First MCP Server in 6 Steps. Towards Data Science, 2025. https://towardsdatascience.com/model-context-protocol-mcp-tutorial-build-your-first-mcp-server-in-6-steps/

[45] Destin Gong. “MCP Client Development with Streamlit: Build Your AI-Powered Web App.” Towards Data Science, 2025. https://towardsdatascience.com/mcp-client-development-with-streamlit-build-your-ai-powered-web-app/

[46] Anna Pawłowska, Piotr Karwat, Norbert Żołek. “Letter to the Editor. Re: [Dataset of breast ultrasound images by W. Al-Dhabyani, M. Gomaa, H. Khaled & A. Fahmy, Data in Brief, 2020, 28, 104863].” Data in Brief, 2023. https://www.sciencedirect.com/science/article/pii/S2352340923003669

7. Glossary

Accuracy : A metric representing the proportion of correct predictions over the total number of predictions.

Confusion Matrix : A table used to evaluate classification performance, showing predicted versus true labels and highlighting correct and incorrect predictions.

CNN (Convolutional Neural Network) : A type of deep learning model particularly effective for image recognition tasks, designed to automatically learn spatial patterns such as edges, shapes, and textures.

Epoch : One complete pass of the entire training dataset through a learning algorithm during model training.

F1-score : A harmonic mean of precision and recall, used to evaluate the balance between false positives and false negatives in classification tasks.

Feedforward Neural Network : A neural network architecture in which data flows in a single direction from input to output, without cycles or feedback loops.

Filter Optimization : The process by which a CNN learns the most effective filters (or kernels) during training to extract relevant visual features from input images.

GPU (Graphics Processing Unit) : A specialized processor initially designed for rendering graphics, now extensively used in AI to accelerate training and inference through large-scale parallel computations.

LLM (Large Language Model) : An artificial intelligence model trained on large amounts of text data, capable of understanding, generating, and reasoning with human-like language (e.g., ChatGPT).

MCP protocol : A standardized communication protocol that allows different software tools and systems to interact seamlessly, supporting modular and reusable AI workflows.

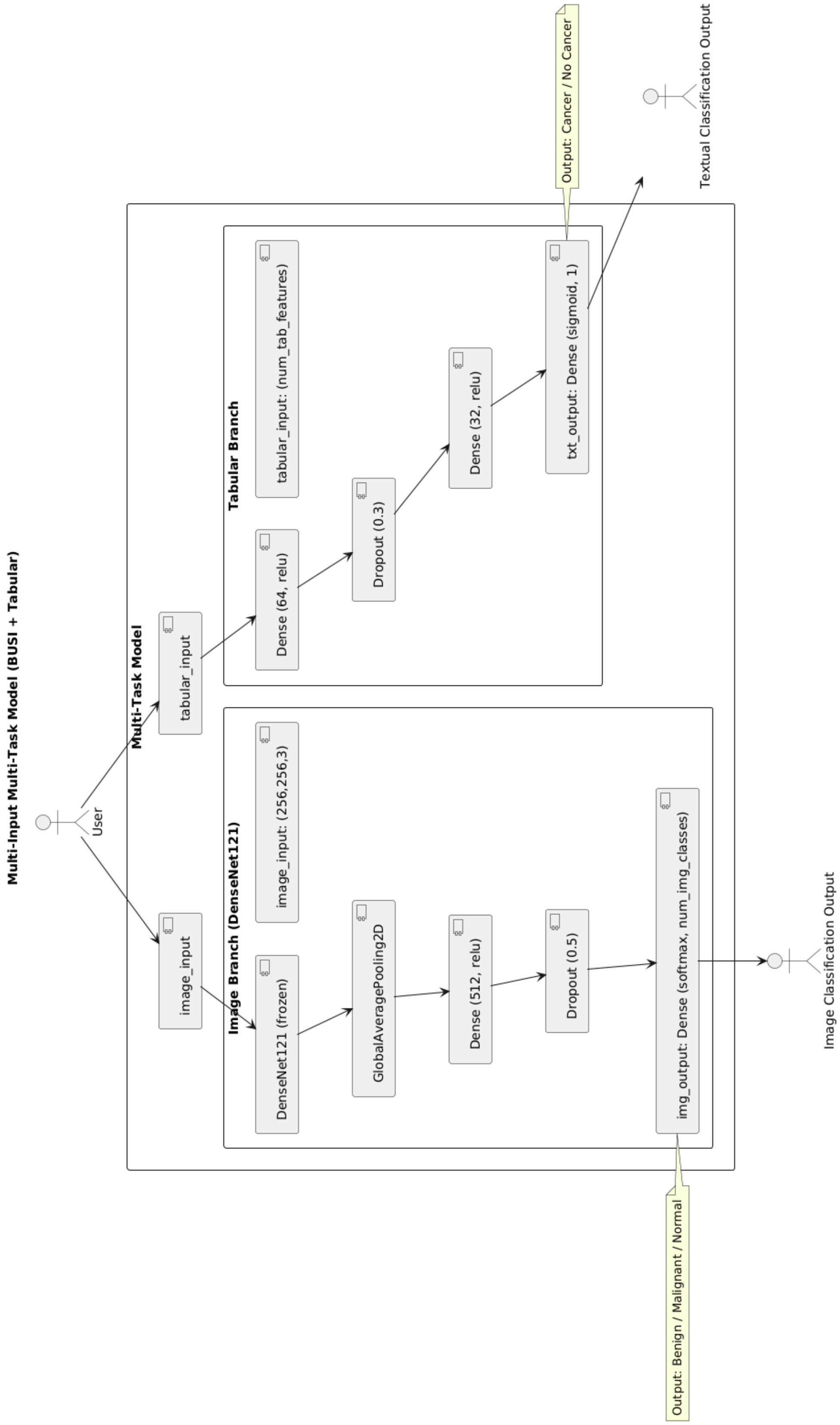
Overfitting: Occurs when a model learns its training data too well, including noise and specific details, to the point where it performs poorly on new, unseen data because it fails to generalize

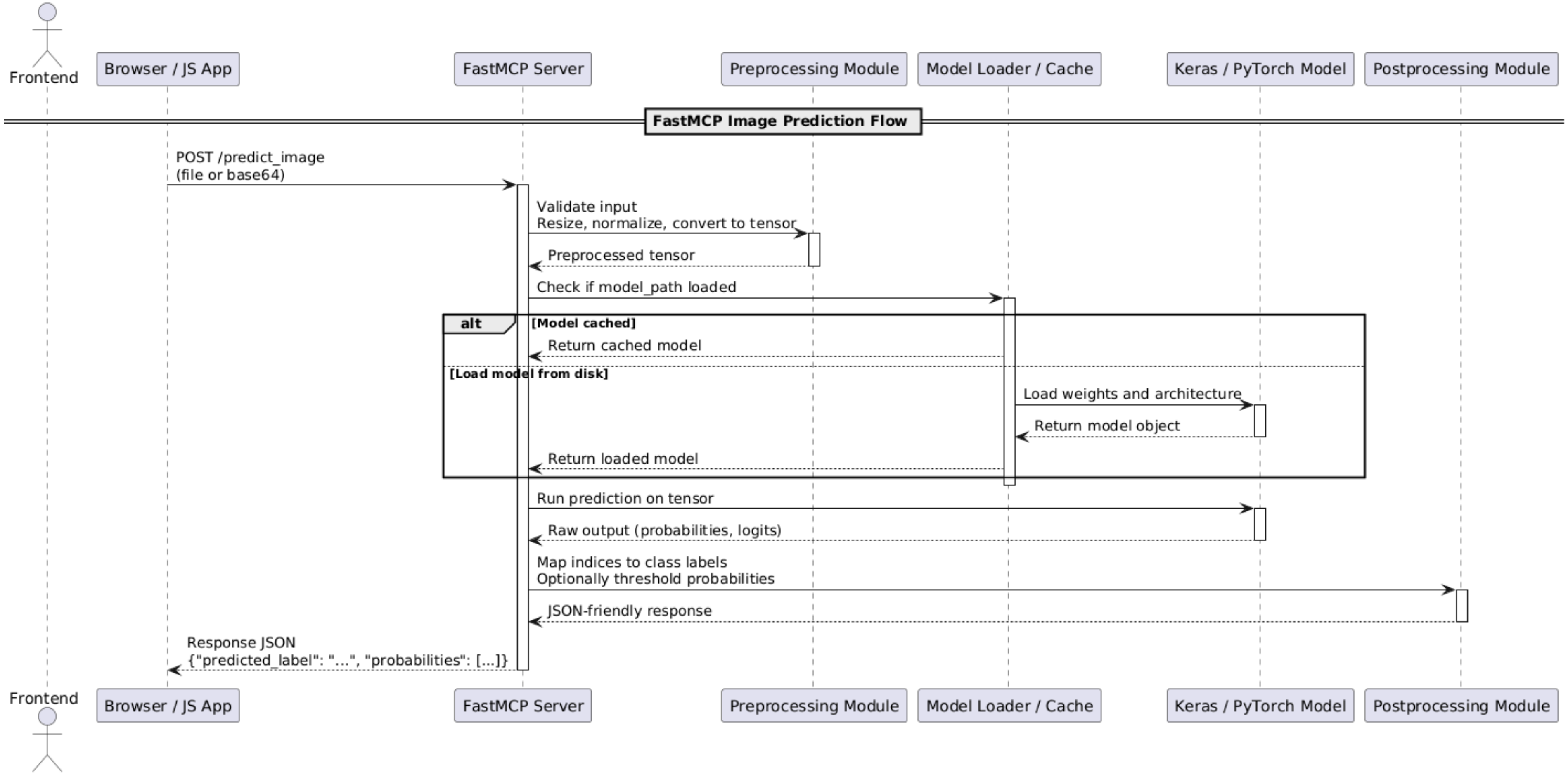
Python : A high-level, versatile programming language widely used in artificial intelligence, data science, and web development, valued for its readability and extensive ecosystem of libraries.

ROC-AUC (Receiver Operating Characteristic – Area Under Curve) : A metric that measures the ability of a classifier to distinguish between classes across all thresholds; higher values indicate better discriminatory performance.

SHAP (SHapley Additive exPlanations) : A technique for interpreting machine learning models by assigning each feature an importance value for a particular prediction, providing both global and local interpretability.

8. Appendices


Multi-Input Multi-Task Model (BUSI + Tabular)
User
Multi-Task Model
image_input
tabular_input
Image Branch (DenseNet121)
image_input: (256,256,3)
DenseNet121 (frozen)
GlobalAveragePooling2D
Dense (512, relu)
Dropout (0.5)
img_output: Dense (softmax, num_img_classes)
Output: Benign / Malignant / Normal
Image Classification Output
Tabular Branch
tabular_input: (num_tab_features)
Dense (64, relu)
Dropout (0.3)
Dense (32, relu)
txt_output: Dense (sigmoid, 1)
Output: Cancer / No Cancer
Textual Classification Output


Frontend
Browser / JS App
FastMCP Server
Preprocessing Module
Model Loader / Cache
Keras / PyTorch Model
Postprocessing Module
FastMCP Image Prediction Flow
POST /predict_image
(file or base64)
Validate input
Resize, normalize, convert to tensor
Preprocessed tensor
Check if model_path loaded
alt
[Model cached]
Return cached model
[Load model from disk]
Load weights and architecture
Return model object
Return loaded model
Run prediction on tensor
Raw output (probabilities, logits)
Map indices to class labels
Optionally threshold probabilities
JSON-friendly response
Response JSON
{"predicted_label": "...", "probabilities": [...]}
Frontend
Browser / JS App
FastMCP Server
Preprocessing Module
Model Loader / Cache
Keras / PyTorch Model
Postprocessing Module