



UNIVERSITÀ
DI PAVIA

UNIVERSITY OF PAVIA
FACULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL, COMPUTER AND BIOMEDICAL ENGINEERING

MASTER'S DEGREE IN COMPUTER ENGINEERING

MASTER THESIS

**Federated Learning for Embryo Evaluation in Assisted
Reproduction:**
development of distributed predictive models for assessing
embryonic development

Candidate: Marco Caretti

Supervisor: Prof. Giovanna Nicora
Co-supervisor: Lorenzo Corso

A.Y. 2026/2027

Abstract

Choosing which embryo to transfer is one of the most delicate steps of an in vitro fertilization (IVF) cycle. It still relies largely on the embryologist’s visual assessment of morphology, which is subjective and hard to reproduce. Time-lapse imaging and machine learning offer a way to describe embryo development quantitatively. However, in order to achieve good predictive performance, a sufficient amount of data need to be collected for training; those data could be spread across clinics and protected by strict regulations such as the GDPR and the EU AI Act. Federated Learning (FL) addresses exactly this situation: it trains a shared model without the data ever leaving the institution that owns them.

In this thesis, FL has been applied to the prediction of blastocyst formation from optical-flow time series extracted from time-lapse videos of embryos development. The dataset contains 5011 embryos acquired between 2013 and 2020. To mimic a realistic multi-center setting, the dataset has been split in 8 different subsets, 7 used for the training of the models and 1 kept for the global evaluation. Five model configurations have been compared: ROCKET, ConvTran, a hybrid LSTM-FCN network, and the MOMENT foundation model, used both with full fine-tuning and as a frozen feature extractor with a small classification head on top. Every model is trained with FedAvg, using the Flower framework, for 2 to 7 clients, and compared with the same model trained on the pooled data.

LSTM-FCN gives the best and most stable results. In the federated setting it reaches an accuracy of 0.84–0.87 on the global test set, matching or beating its centralized counterpart (0.83–0.84). The MOMENT-embedding model reaches about 0.81 with seven clients while exchanging only the classification head, which keeps the communication cost very low. A repeated-measures ANOVA shows a strong effect of the architecture ($p < 0.001$) and a significant interaction between architecture and training mode ($p = 0.030$), whereas the training mode alone has no significant effect ($p = 0.093$).

Finally, a federated explainability procedure based on SHAP is proposed, which is called SHAP-FL. Each client computes Shapley values locally on its own data and sends the server only an aggregated importance vector, which the server combines into a global explanation. Applied to LSTM-FCN, SHAP-FL, shows that the model bases its decision on the last time period of culture, when cavitation and blastocyst expansion take place. the explanations of the seven centers agree closely (mean Spearman correlation 0.95).

Contents

Abstract	i
1 Introduction	1
1.1 Artificial intelligence in embryology	2
1.2 The Generalizability and privacy problem	2
1.3 The need for Explainability in the clinical domain	4
1.4 Objectives	4
1.5 Thesis structure	5
2 Clinical Background	7
2.1 IVF workflow	7
2.1.1 From stimulation to gamete retrieval	7
2.1.2 Fertilization and embryo culture	8
2.1.3 The blastocyst stage	8
2.2 Human morphological assessment	9
2.2.1 The Gardner system	10
2.2.2 Limits of visual inspection	11
3 Existing decision-support systems	13
3.1 Looking at an embryo	13
3.2 Commercial systems	13
3.2.1 iDAScore	14
3.2.2 CHLOE	14
3.3 Research frameworks	15
3.3.1 BlastAssist	15
3.3.2 EmbryoMLSelection	15
3.4 CellPIV	16
3.4.1 The CellPIV software	16
3.4.2 The PIV algorithm	16
3.4.3 Derived metrics	17
3.4.4 Strengths and limitations	18
3.5 Comparison and limitations of these systems	19

4	Federated Learning	21
4.1	Problem formulation	21
4.2	Aggregation strategies	22
4.3	Federated Averaging	23
4.4	Heterogeneous (non-IID) data	23
4.5	Privacy	24
4.6	The Flower framework	25
5	Explainable AI in a federated setting	27
5.1	Why explanations matter	27
5.2	Explanation methods	28
5.3	Shapley values	28
5.4	SHAP: SHapley Additive exPlanations	29
5.4.1	Additive feature attribution	29
5.4.2	The background dataset	29
5.4.3	Estimating SHAP values	30
5.4.4	From local to global explanations	30
5.5	Explainability in Federated Learning	30
5.5.1	Challenges	31
5.6	How SHAP-FL works	31
5.6.1	Properties	32
5.6.2	Measuring agreement between explanations	32
5.7	Explanation level	33
6	Dataset and data preparation	35
6.1	Origin and content	35
6.2	Optical-flow time series	36
6.3	Data quality and preprocessing	36
6.4	Simulating a federated environment	37
6.4.1	How many partitions? A bootstrap analysis	37
6.4.2	Partitioning	38
7	Time series classification models	41
7.1	ROCKET	41
7.2	ConvTran	42
7.3	LSTM-FCN	43
7.4	MOMENT	44
7.4.1	Architecture and pre-training	44
7.4.2	Option 1: full fine-tuning	44
7.4.3	Option 2: frozen embeddings and a classification head	44
7.5	Summary	45

8	Implementation	47
8.1	Software	47
8.1.1	ServerApp	47
8.1.2	ClientApp	48
8.2	Experimental protocol	48
8.3	Evaluation metrics	49
8.4	Statistical analysis	49
8.5	SHAP-FL configuration	50
9	Results	51
9.1	Federated vs centralized training	51
9.2	Metrics	54
9.3	Statistical analysis	55
9.3.1	Repeated-measures ANOVA	55
9.3.2	Per-model comparisons	56
9.4	Discussion of performance	56
9.4.1	LSTM-FCN works best	56
9.4.2	Model size and stability	57
9.4.3	Embedding approach	57
9.5	Federated explainability results	58
9.5.1	Global importance	58
9.5.2	Agreement between clients	59
9.5.3	Limits	60
10	Conclusions and future work	61
10.1	Summary	61
10.2	Answers to the research questions	61
10.3	Limitations	62
10.4	Future work	63
10.5	Final remarks	64
	Bibliography	65

List of Figures

1.1	Example of Time-Lapse Imaging data: frames of two embryo from the dataset, one that reached the blastocyst stage and one that not reached that stage. Figure taken from [1]	3
1.2	Graphical abstract of the thesis. (1) Optical-flow time series are extracted from time-lapse videos and split by acquisition year into seven federated clients and an independent global test set. (2) Five model configurations are considered. (3) Each model is trained with Federated Learning (FedAvg, Flower). (4) All models are evaluated on the global test set, and the federated predictions are explained with SHAP-FL.	5
2.1	The main steps of an IVF treatment. (1) Ovarian stimulation with gonadotropins induces the growth of several follicles. (2) The oocytes are collected by ultrasound-guided aspiration of the follicles. (3) They are fertilized in the laboratory, by conventional IVF or by intracytoplasmic sperm injection (ICSI, shown here). (4) The embryos are cultured for up to five or six days, possibly in a time-lapse incubator that records images at regular intervals. (5) The embryo with the highest estimated potential is transferred into the uterus. (6) If development proceeds, the blastocyst hatches from the zona pellucida and implants into the endometrium. Image generated with Claude	9
2.2	Schematic representation of the blastocyst grading system introduced by Gardner and Schoolcraft. top : degree of expansion. bottom : quality of the inner cell (left) and of the trophectoderm (right). Image generated with Claude	11
3.1	Principle of Particle Image Velocimetry as implemented in CellPIV. Each interrogation region of frame t is cross-correlated with the corresponding area of frame $t + \Delta t$; the position of the correlation peak gives the local displacement, and repeating the procedure over all regions yields the velocity field. Based on the scheme in [2]. . . .	17

5.1	The SHAP-FL procedure. Each client explains the global model on its own data with a local background and shares only its aggregated importance vector; the server computes the global importance as a weighted average. Image generated with Claude	32
6.1	Median (line) and interquartile range (shaded area) of the optical-flow series for embryos that do (blue) or do not (orange) reach the blastocyst stage, computed on the training partitions. The dotted line marks the last time point fed to the models.	37
6.2	Average accuracy and F1-score as a function of the number of partitions, from the bootstrap analysis. More partitions mean fewer samples per node; beyond 8 partitions the metrics drop markedly. .	38
6.3	Number of embryos and percentage of blastocysts per partition. . .	39
7.1	The LSTM-FCN variant used in this work.	43
7.2	The MOMENT-embedding pipeline: the pre-trained encoder is frozen and run locally; only the classification head takes part in federated training.	45
8.1	Experimental protocol: for each $N \in \{2, \dots, 7\}$ the corresponding partitions are used to train the model in federated and centralized mode; both models are evaluated on the global test set.	48
9.1	Accuracy on the global test set as a function of the number of partitions, for each model, in federated (solid line) and centralized (dashed line) training.	52
9.2	Accuracy difference between federated and centralized training for each model and number of partitions. Positive values (blue) indicate higher accuracy for federated training.	53
9.3	Confusion matrices on the global test set ($N = 7$) for LSTM-FCN and MOMENT-embedding in both modes (0 = no blastocyst, 1 = blastocyst).	54
9.4	Matthews correlation coefficient on the 2020 test set as a function of the number of partitions.	55
9.5	Accuracy on the global test set of each architecture in federated (blue) and centralized (orange) mode. Dots are the individual runs. Architectures are ordered by their Friedman mean rank in federated mode (1 = best in every configuration); the mean ranks of both modes are given below each name.	57

9.6	Federated global importance of the input time points for the LSTM-FCN model. (a) Importance $\bar{I}_j$ of each time point (thin line) and its two-hour moving average (thick line); the shaded bands mark the days of culture. (b) Share of the total importance falling in each day of culture; the error bars give the range across the seven clients, and the black marks the share each day would have if the importance were uniform over time.	58
9.7	Mean signed attribution of each time point (two-hour moving average) for the explained embryos that do (blue) and do not (orange) reach the blastocyst stage over the seven clients. Positive values push prediction towards blastocyst formation.	59
9.8	Agreement between the local explanations produced by SHAP-FL on the seven clients. (a) Spearman correlation between the local importance vectors of the time points. (b) Local importance of each time point (two-hour moving average), normalized on the total of each client.	60

List of Tables

3.1	Summary comparison of the decision-support systems discussed in this chapter.	20
4.1	Main differences between centralized and federated learning. . . .	24
6.1	Composition of the partitions. Partitions 0–6 are the federated clients; 2020 is the global test set.	39
7.1	Model characteristics and federated parameters.	45
9.1	Accuracy on the global test set as the number of partitions N varies. F = federated (FedAvg), C = centralized, FT = Fine Tuning, emb = Embeddings. The best value in each row and mode is in bold. . . .	51
9.2	Repeated-measures ANOVA on accuracy (subject: number of partitions, $n = 6$).	56
9.3	Mean accuracy difference (federated – centralized) and Wilcoxon test for each model.	56

Chapter 1

Introduction

Infertility has become a major public health issue. The World Health Organization estimates that it affects roughly 17.5% of the adult population worldwide [3, 4]. Assisted Reproductive Technology (ART), and in vitro fertilization (IVF), has changed reproductive medicine profoundly and allowed millions of couples to overcome obstacles that used to be insurmountable. Yet, despite decades of technical progress since the first successful IVF procedures in the 1970s, the success rate per treatment cycle is still modest. Measured as live births, it averages between 25% and 35%, and it varies a lot with maternal age and gamete quality [5, 6].

One of the most critical steps in the whole process, is deciding which embryo to transfer to the uterus [7]. A typical stimulation cycle yields several oocytes; once fertilized, embryos are cultured in vitro for three to six days, up to the blastocyst stage. The embryo(s) with the best chance of implanting have traditionally been chosen through a qualitative or semi-quantitative morphological assessment [8], carried out by experienced embryologists who look at the embryos under the microscope at a few fixed points in time. This approach has three well-known weaknesses. Firstly, subjectivity: different embryologists, and sometimes the same embryologist on different days, read the morphological criteria differently [9–11]. Secondly, a handful of snapshots cannot capture what is in fact a continuous, dynamic process [12]. Thirdly, what we can see of an embryo does not always reflect its chromosomal make-up or its metabolic potential [13].

To raise the chance of pregnancy, clinics used to transfer more than one embryo at a time. This led to many more multiple pregnancies, which carry real risks for both mother and fetus [14]. The medical community now strongly recommends single embryo transfer (SET) [15]; being able to pick the single most viable embryo has therefore become a research priority.

A better explanation of the IVF workflow will be provided in Chapter 2

1.1 Artificial intelligence in embryology

In recent years, much research has focused on digitalization and automated embryo assessment, to overcome the limits of manual evaluation and make SET more effective [16]. Time-lapse imaging (TLI) [17] has been a turning point: incubators equipped with a camera record the embryo continuously, at high resolution, without taking it out of its culture environment (??). The result is an unprecedented amount of morphokinetic data.

Artificial intelligence (AI), and machine learning (ML) have quickly become central in this area [18]. Deep neural networks can pick up patterns in images and videos that a human observer simply cannot see. Used well, they make embryo scoring more objective, reducing human error, and predicting both implantation and euploidy non-invasively [19]. Systems such as BlastAssist [20], EmbryoMLSelection [21] and other recent architectures [22] show how far the meeting of data engineering and reproductive medicine has already gone.

More recently, the field of AI has been reshaped by the advent of Foundation Models [23]. These are large neural networks that are pre-trained on a vast and diverse collection of data. During pre-training they learn general-purpose representations, which can be adapted to many tasks either by fine-tuning the whole model or simply by using its internal representation as features. After transforming natural language processing and computer vision, this paradigm has recently been extended to time series with models such as MOMENT [24]. Those are pre-trained on large corpora of series from heterogeneous domain. In a clinical setting like ours, where labeled data are difficult and expensive to collect, this is really interesting: a model that has already learned what time series "look like" in general can reach good performance with fewer examples than one trained from scratch.

A better explanation of the existing AI-based decision-support system existing will be provided in Chapter 3

1.2 The Generalizability and privacy problem

Deep learning models generalize well only when they are trained on large and varied datasets. However, ART data are scattered across many clinics, each with a limited number of cases, its own laboratory protocols, and its own patient population. The combination of these data in one place is difficult due to strict regulations: the General Data Protection Regulation (GDPR) [25, 26], the rules on medical devices [27], and the recent European AI Act [28]. In practice, then, most models end up being trained on data from a single clinic, and they risk fitting the local population and acquisition conditions too closely to work elsewhere.

Federated Learning (FL) [29, 30] tackles this problem head on. Instead of moving the data to the model, it moves the model to the data. Each participant, called client (in our case, a clinic), trains a local copy on its own data and sends a coordinating

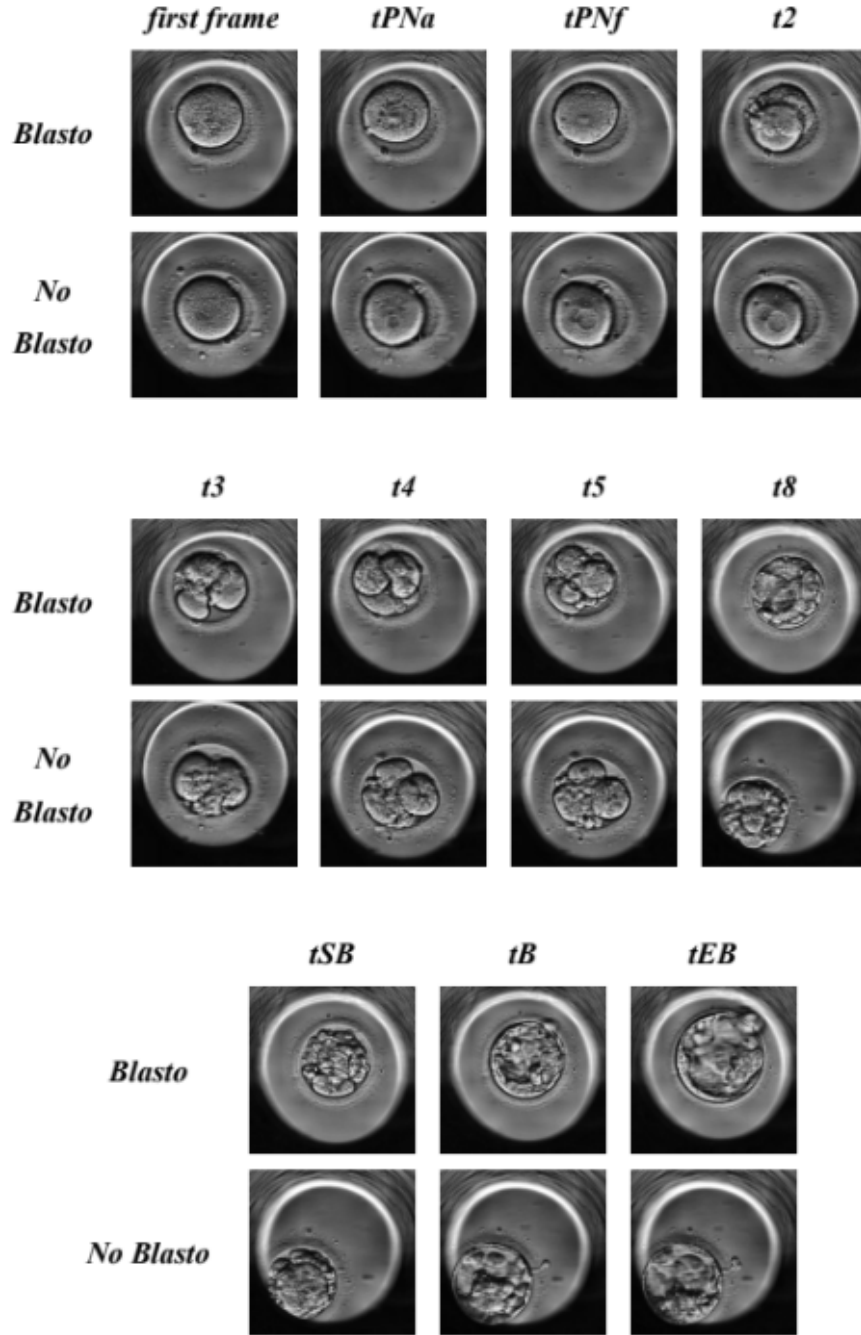


Figure 1.1: Example of Time-Lapse Imaging data: frames of two embryo from the dataset, one that reached the blastocyst stage and one that not reached that stage. Figure taken from [1]

server only the updated parameters, which the server then merges into a global model. Several institutions can thus contribute to the same model while sensitive data remain where they are [31, 32].

A better explanation of Federated Learning will be provided in Chapter 4

1.3 The need for Explainability in the clinical domain

In a clinical setting, a model that is accurate but "black-box" is not enough. A decision-support system has to be able to justify its outputs: before relying on it, an embryologist needs to understand why a given embryo is considered promising [13, 33]. Deep networks are essentially black boxes, and using them for something as consequential as embryo selection raises questions of transparency, accountability and regulatory compliance. Explainable AI (XAI) [34] provides tools to trace each prediction back to the factors that produced it. Among them, methods based on Shapley values, such as SHAP [35], have particularly solid theoretical foundations.

Explaining a federated model, is harder than explaining a centralized one. Post-hoc methods like SHAP need access to reference data and to the samples being explained, and in FL those data, by design, never leave the clients. We therefore need a way to produce both local and global explanations without breaking the rule that data stay local.

A better explanation of SHAP values and how they works in a Federated Environment will be provided in Chapter 5

1.4 Objectives

The overall goal of this thesis is to design and evaluate a federated and explainable approach for predicting blastocyst formation from optical-flow time series extracted from time-lapse videos. More specifically, we set out to:

1. build a realistic federated simulation in which the data are split, so that clients differ from one another in a non-IID (non-Independent and Identically Distributed) way, with one whole year held out as an independent global test set;
2. compare different time series classifiers (ROCKET, ConvTran, LSTM-FCN and the MOMENT foundation model) in federated and centralized training, for a growing number of clients;
3. evaluate whether the foundation model (MOMENT) can be used as a frozen feature extractor, where only a small classification head is federated, which cuts communication and client-side computation dramatically;
4. test whether models trained with federated learning underperform compared to models trained in the traditional way;

5. design and apply a federated explainability procedure based on SHAP (SHAP-FL), which yields importance scores for each client and an aggregated global importance without sharing raw data and use it to study how consistent the explanations are across centers.

Figure 1.2 gives an overview of the whole workflow, from the time-lapse videos to the federated explanation.

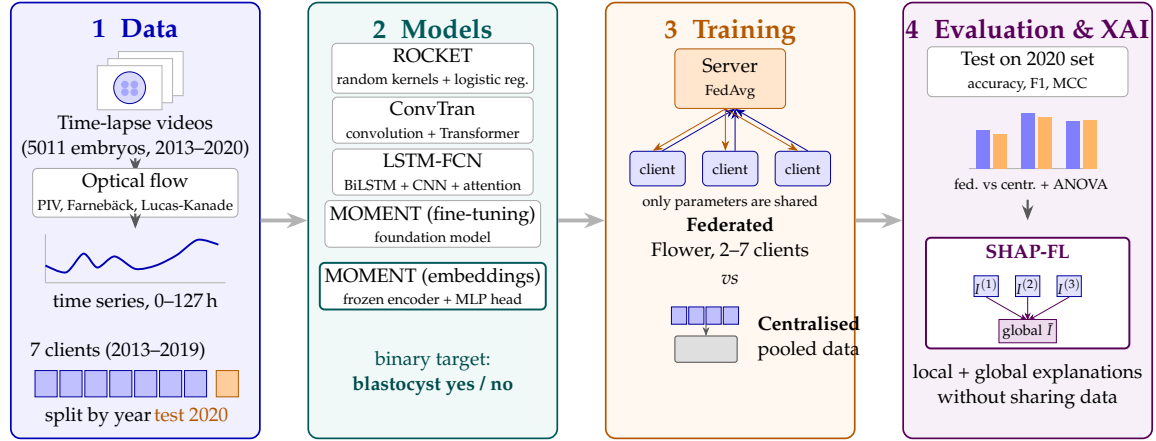


Figure 1.2: Graphical abstract of the thesis. (1) Optical-flow time series are extracted from time-lapse videos and split by acquisition year into seven federated clients and an independent global test set. (2) Five model configurations are considered. (3) Each model is trained with Federated Learning (FedAvg, Flower). (4) All models are evaluated on the global test set, and the federated predictions are explained with SHAP-FL.

1.5 Thesis structure

Chapter 2 describes the IVF workflow, the main stages of embryo development, the Gardner grading system and the limits of visual inspection. Chapter 3 reviews existing AI-based decision-support systems for embryo selection. Chapter 4 introduces Federated Learning, the main aggregation algorithms, and the privacy and regulatory aspects. Chapter 5 covers explainable AI, from Shapley values to SHAP and its use in a federated setting, and defines SHAP-FL. Chapter 6 presents the dataset, the extraction of optical-flow series, the quality checks and the partitioning strategy. Chapter 7 describes the classification models, and Chapter 8 the federated implementation, the experimental protocol, the metrics and the statistical tests. Chapter 9 reports and discusses the results. Finally, Chapter 10 draws the conclusions, discusses the limitations of this work and outlines future directions.

Chapter 2

Clinical Background

This chapter covers the biological and clinical background needed to follow the rest of the thesis. Firstly, the thesis goes through the steps of an IVF cycle and the main stages of preimplantation development; secondly, it looks at how embryologists judge embryo quality. This is important for understanding both, what our models are asked to predict (whether a blastocyst forms), and what kind of signal they receive as input.

2.1 IVF workflow

IVF is a long chain of medical, surgical and laboratory procedures that have to be carefully coordinated. The goal is to fertilize gametes outside the body, guide the early development of the resulting embryos in a tightly controlled environment, and finally identify the embryo with the highest biological potential so that it can be transferred to the uterus [5].

2.1.1 From stimulation to gamete retrieval

IVF starts with Controlled Ovarian Stimulation (COS) [36]. In a natural cycle, the endocrine system selects a single dominant follicle and brings it to maturity. In the COS procedure, exogenous hormones (gonadotropins such as FSH (Follicle-Stimulating Hormone) and LH (Luteinizing Hormone)) are given, so that a whole group of follicles develops at the same time and the chances of success increase. The ovarian response is followed with transvaginal ultrasound and repeated blood tests. Once the follicles are large enough, a hormonal trigger, usually hCG (Human Chorionic Gonadotropin) or a GnRH (Gonadotropin-Releasing Hormone) agonist [37], is given to mimic the natural LH surge. This lets the oocytes resume meiosis and complete their maturation up to metaphase II (MII).

About 34–36 hours after the trigger, the oocytes are retrieved. Under ultrasound guidance and in sterile conditions, the follicles are aspirated through the vaginal wall [38], and the follicular fluid goes straight to the embryology lab, where the

cumulus-oocyte complexes (COCs) are isolated. Meanwhile, the semen sample is prepared. Washing and selection techniques such as density gradients or swim-up separate the sperm from the seminal plasma and cell debris, keeping the fraction with the best progressive motility and morphology [39, 40].

2.1.2 Fertilization and embryo culture

With the gametes ready, the next step is insemination. Depending on the semen quality and the couple's history, fertilization is achieved either through conventional IVF, where oocytes and sperm are simply placed together in a drop of culture medium and the sperm penetrates the oocyte on its own; or through intracytoplasmic sperm injection (ICSI). ICSI is essential in cases of severe male-factor infertility: the cumulus cells are removed (denudation) and a single selected sperm cell is injected directly into the cytoplasm of the MII oocyte [41].

Around 16–18 hours later (day 1), the embryologists check whether fertilization has occurred by looking for two pronuclei (2PN, one maternal and one paternal) and two polar bodies [42]. From then on, the zygote develops inside modern incubators that keep temperature, pH and gas concentrations strictly constant, so as to reproduce the conditions of the oviduct and uterus [43].

On days 2 and 3, the embryo goes through a series of mitotic divisions without growing in overall size, splitting into smaller and smaller cells called blastomeres (the cleavage stage [44]). Around day 4 it becomes a morula: the cells bind tightly to one another through tight junctions, a process called compaction, and their boundaries become hard to make out under the microscope [45].

2.1.3 The blastocyst stage

Between day 5 and day 6 the embryo undergoes a key transformation and becomes a blastocyst. Fluid starts to collect inside it and forms a central cavity, the blastocoel. This leads to the first cell differentiation of embryonic life, with two distinct populations: the **inner cell mass** (ICM), a compact cluster of pluripotent cells [46, 47], and the **trophoblast** (TE), a single layer of epithelial cells that surrounds the cavity and will form the placenta and the other extra-embryonic tissues [48].

At day 2 or 3 the embryo reach the Embryonic Genome Activation (EGA). Many embryos with chromosomal abnormalities or metabolic defects stop developing at this point and never form an expanded blastocyst, so extended culture to this stage works as a first, natural biological filter [49]. Whether an embryo makes it to the blastocyst stage is therefore a meaningful indicator of its developmental potential, and it is exactly what the models in this thesis try to predict.

The workflow ends with embryo evaluation and transfer. The embryo with the best developmental potential is loaded into a thin catheter and gently placed in the patient's uterine cavity. Any other good-quality blastocysts are vitrified (ultra-rapid

cryoconservation) and kept for later attempts [50]. Figure 2.1 shows each step of the workflow.

It is precisely during this long and dynamic culture window, and especially in the fast transition from morula to expanded blastocyst, that human observation shows its limits. This is where continuous time-lapse monitoring and AI algorithms can make a difference [19].

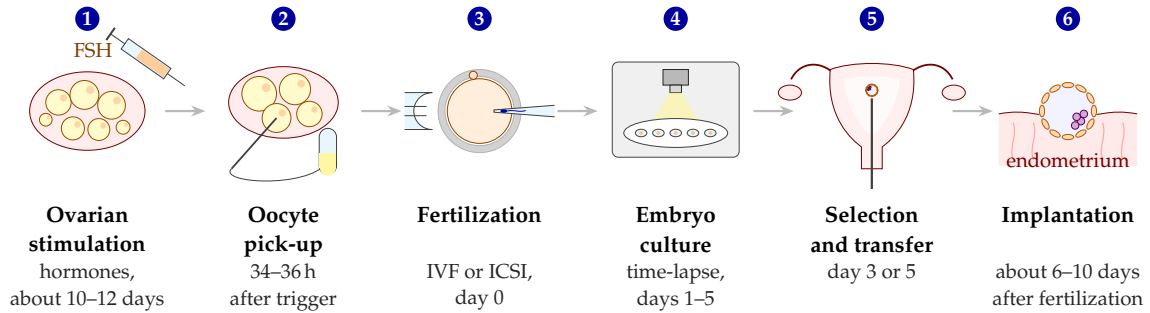


Figure 2.1: The main steps of an IVF treatment. (1) Ovarian stimulation with gonadotropins induces the growth of several follicles. (2) The oocytes are collected by ultrasound-guided aspiration of the follicles. (3) They are fertilized in the laboratory, by conventional IVF or by intracytoplasmic sperm injection (ICSI, shown here). (4) The embryos are cultured for up to five or six days, possibly in a time-lapse incubator that records images at regular intervals. (5) The embryo with the highest estimated potential is transferred into the uterus. (6) If development proceeds, the blastocyst hatches from the zona pellucida and implants into the endometrium. Image generated with Claude

2.2 Human morphological assessment

Despite progress in omics and digital monitoring, direct visual observation has been the clinical gold standard for decades. Embryologists use it to judge embryo quality and to decide which embryos to transfer or freeze. They inspect the embryos under the microscope at conventional moments in time, using criteria that depend on the developmental stage.

During the first 72 hours after insemination, which correspond to the cleavage stage, attention goes to how the first divisions unfold. The embryologist counts the blastomeres, checks whether they are of similar size, and looks for cytoplasmic fragmentation and multinucleation. Under good conditions, an embryo has two to four cells 48 hours after fertilization and six to eight cells by 72 hours.

Each cell division consists of two closely linked steps: mitosis, in which the duplicated chromosomes are separated into two nuclei; and cytokinesis, in which the cytoplasm itself is split and the cell physically divides in two. During cytokinesis a contractile ring of actin and myosin forms beneath the cell membrane and gradually tightens, pinching the cell until two daughter cells are separated. When

this process goes wrong, the cytoplasm may not be partitioned cleanly, and small portions of it can bud off from the blastomeres. These fragments of cytoplasm without a nucleus are known as cytoplasmic fragments [44]. The embryologist estimates what share of the embryo's volume these fragments take up and grades the embryo accordingly, from excellent (no fragments, or less than 10%) down to severely compromised (more than 50%). Blastomeres of even size and the absence of cells with several nuclei are further signs that mitosis and cytokinesis, and with them chromosome segregation, have gone well.

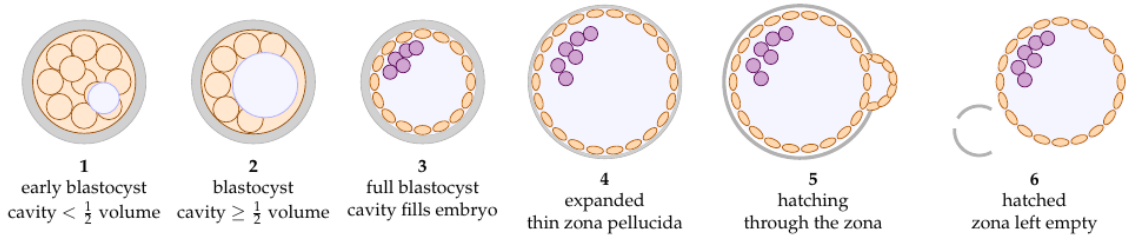
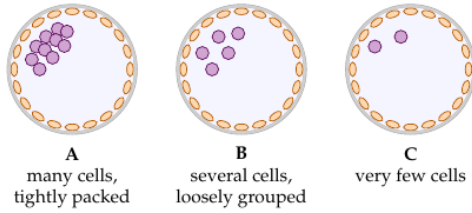
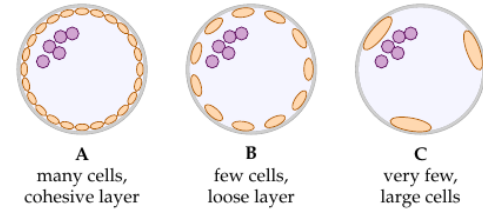
2.2.1 The Gardner system

When culture is extended to day 5 or 6, the embryo reorganizes completely into a blastocyst. To assess this three-dimensional structure, clinics almost universally use the grading system introduced by Gardner and Schoolcraft back in 1999 [47]. It gives each blastocyst a code made of one number and two letters, each describing one aspect of its morphology.

The number, from 1 to 6, describes how expanded the blastocyst is and how much the zona pellucida has thinned under the pressure of the fluid in the blastocoel. At grade 1 the cavity is barely visible and takes up less than half of the embryo; at grade 2 it occupies half of the volume or more; at grades 3 and 4 the blastocoel fills the embryo completely or has clearly expanded it. Grades 5 and 6 describe the final stage, when the blastocyst starts to come out of its shell (hatching) or has come out completely.

The first letter grades the ICM, the cells that will eventually form the fetus. Grade A means a compact, well-defined mass with many tightly packed cells; grade B a looser, less dense group; grade C very few cells or clearly disorganized ones. The second letter grades the trophectoderm, which will become the placenta. Grade A is a continuous, uniform layer of many cohesive cells; grade B a less cohesive layer with unevenly distributed cells; grade C a sparse layer of a few large cells. For example, a blastocyst graded "4AA", is fully expanded with an excellent ICM and trophectoderm, and would normally be the first choice for transfer.

Figure 2.2 shows a schematic representation of the appearance of the embryo at different gradings.

Expansion (1–6)**Inner cell mass (ICM)****Trophoblast (TE)**

● zona pellucida ● trophoblast ● inner cell mass ● blastocoel (cavity)

Figure 2.2: Schematic representation of the blastocyst grading system introduced by Gardner and Schoolcraft. **top:** degree of expansion. **bottom:** quality of the inner cell (left) and of the trophoblast (right). Image generated with Claude

2.2.2 Limits of visual inspection

Widely used as it is, visual assessment has some built-in weaknesses that limit how well it predicts outcomes.

Firstly, subjectivity: several studies have shown that different embryologists, or even the same embryologist at different times, can give the same embryo different grades, especially in the intermediate quality classes [9–11].

Secondly, traditional observation is static: to look at an embryo, the dish has to be taken out of the incubator for a moment, so development can only be followed through occasional snapshots. Fast or transient events, such as the actual speed of compaction or brief collapses of the blastocoel, are easily missed [12].

Finally, morphology says little about the chromosomes. A sizeable share of blastocysts rated excellent under the Gardner system turn out to be aneuploid, which leads to implantation failure or early miscarriage [13]. Because visual inspection cannot see this deeper biology, clinical judgment needs to be complemented by objective, continuous and quantitative tools. This is the gap that AI models working on time-lapse data aim to fill.

Chapter 3

Existing decision-support systems

Bringing AI into reproductive medicine is about turning embryo selection from a qualitative, subjective judgment into a data-driven analysis. Time-lapse systems provided the infrastructure for this shift, since they produce large streams of high-resolution images and videos that cover the whole preimplantation period [17]. Machine learning and deep learning have become the standard tools for extracting the complex, non-linear patterns hidden in these data [16, 19].

This chapter reviews the main systems that have been proposed. Firstly, different algorithms to face this problem will be outlined. Secondly, two commercial systems already used in clinical practice (CHLOE and iDAScore) and two research frameworks (BlastAssist and EmbryoMLSelection) will be described. Finally, CellPIV will be described more in detail since it is essential to understand how the input data of this thesis have been extracted.

3.1 Looking at an embryo

AI applications in embryology follow one of three strategies. Some analyze static images, usually of the blastocyst on day 5 or 6, and try to reproduce or improve on the embryologist's morphological grading. Others focus on morphokinetics, exploiting the time dimension of time-lapse recordings, either through annotated division times or by feeding the whole video to a deep network. Finally, another strategy is to quantify cytoplasmic movement, measuring tiny motions inside the cell with techniques borrowed from fluid dynamics.

3.2 Commercial systems

Over the last few years, several AI tools for embryo assessment have reached the market, usually integrated with a specific time-lapse incubator. Their internal models and training data are proprietary, so they cannot be reproduced or retrained, but they are important because they show what is already being used in clinical practice and how such systems are validated.

3.2.1 iDAScore

iDAScore (intelligent Data Analysis Score), developed by Vitrolife [51] for the EmbryoScope time-lapse incubators, is a fully automated deep learning model that scores embryos directly from their time-lapse image sequences, with no manual annotation [52]. Given the image sequence of an embryo, it outputs a score that reflects its likelihood of implantation, which can be used to rank the embryos of a cohort. Scores given by this framework were found to correlate with conventional blastocyst grades.

The first version has been trained on 115 832 embryos from 18 clinics, 14 644 of which had been transferred and had a known implantation outcome (Known Implantation Data, KID). On these transferred embryos, the model reached an AUC of 0.67; on clinics not used for training, the KID AUC ranged from 0.60 to 0.75 [52]. A second version extended the model to embryos transferred on days 2, 3 and 5, using 141 428 embryos from 22 clinics and reached a KID AUC between 0.62 and 0.70 depending on the day of transfer [53].

iDAScore has been evaluated by different clinical groups on retrospective single-clinic studies.

Firstly, a clinic in Japan tested it on 3969 embryos reaching an AUC ranging between 0.66 and 0.72 depending on patient's age [54]. Secondly, in a cohort of 7786 day-3 embryos it reached an AUC around 0.62 [55]. Thirdly, on 3604 blastocysts in Italy, it reached an AUC of 0.60 for euploidy and 0.66 for live birth,

comparable to embryologists' performance [56]. Finally, a study across centers reported an AUC of 0.63 for day 3 and 0.59 for day 5 transfers [57].

None of these studies reported a collapse in performance.

Furthermore, differences between clinics have been studied [58], evaluating the model on 4805 blastocyst transfers from four different clinics and reaching a clinic-specific AUC ranging between 0.58 and 0.69; standardizing for maternal age also reduced between-clinic variance by 16% [58]. Finally, prospective evidences come from a randomized trial conducted on 14 clinics on 1066 patients reaching a pregnancy rate of 46.5% against the 48.2% of standard morphology not demonstrating the non-inferiority of the model, but reducing the time needed for evaluation [59]

3.2.2 CHLOE

CHLOE, developed by Fairtility [60], is an AI-based decision-support platform for IVF laboratories. According to the manufacturer, it estimates the likelihood of blastulation, implantation and ploidy, automatically annotates morphokinetic events and pronuclear count, and ranks the embryos of a cohort. This framework uses segmentation networks (Mask R-CNN with a ResNet-50 backbone) applied to three focal planes of EmbryoScope time-lapse videos to segment the trophoctoderm and the inner cell mass. From the segmented structures it derives morphometric parameters such as blastocyst size, ICM size, the ICM-to-blastocyst ratio and ICM

shape. Compared with iDAScore, CHLOE places more emphasis on interpretable, measurable features, although the full system remains proprietary.

The published evidence on CHLOE is much thinner than for iDAScore. In a retrospective study on 6024 embryos, CHLOE quality score has been used as a measurement tool to study how clinical variables affect development [61]. More detail are available on CHLOE Blast, a module that predicts blastocyst formation by day 5 [62]. The clinical study included 703 embryos from three US sites [63] reaching an AUC of 0.74.

3.3 Research frameworks

3.3.1 BlastAssist

BlastAssist [20] has been designed to measure interpretable features of human embryos automatically and objectively, focusing on segmenting and grading blastocysts on days 5 and 6 from high-resolution images. Its pipeline follows the same logical steps an embryologist would.

Firstly, the images are cropped automatically to isolate the embryo from the background, and color and contrast are normalized. Secondly, convolutional segmentation networks based on the U-Net architecture identify the key structures of the embryo: the zona pellucida, the inner cell mass and the trophectoderm. Finally, quantitative features such as the degree of expansion and the quality of the ICM and TE are computed from the segmented structures: consistent with the Gardner criteria, and turned into a continuous score that can be used to rank embryos.

The pipeline has been developed on more than 67 million images from 32 939 embryos. Its measurements were comparable to those of human experts, for example, an AUROC between 0.84 and 0.94 for fertilization status; a correlation between 0.71 and 0.77 with expert symmetry scores; and an accuracy of 90.0% for developmental timing, against 91.4% for experts. Some of the measured features, such as the time to the 2-cell stage and 2-cell symmetry, were also significantly associated with implantation in 723 single embryo transfers [20].

The strength of BlastAssist is its interpretability. Because it shows the segmented structures, the clinician can see which regions the algorithm based its assessment on. However, all the data comes from a single center and the author themselves note that the pipeline has not been tested on data from other clinics [20], so the generalization of the results is limited.

3.3.2 EmbryoMLSelection

EmbryoMLSelection [21] exploits the time dimension of the time-lapse data. The underlying idea is that how fast and how regularly an embryo moves through its developmental stages says more than its final appearance alone. The framework extracts morphokinetic parameters of early development, such as the timing of

cell divisions and stage transitions; then it combines them with machine learning models to identify a “signature” associated with blastocyst development. It can also include clinical variables such as maternal age and look at several clinical endpoints, making this kind of approach quite close to everyday laboratory practice.

The study included 575 embryos; from 30 candidate variables, the framework selects features, extracts decision rules and evaluates them. The final signature consists of six rules based on eight variables including maternal age, the time of pronuclear fading and the durations of the second and third cell cycles; it reached an AUC of 0.84 and an accuracy of 81% in predicting blastocyst development on a validation cohort of 81 embryos [21]. However, the validation embryos come from the same unit so we have no information on how EmbrioMLSelection generalize.

3.4 CellPIV

3.4.1 The CellPIV software

CellPIV is a collection of MATLAB tools written by Shane Windsor for the analysis of cytoplasmic movement within whole cells. The software works on sequences of 8-bit greyscale images and is organized in three programs, each building on the output of the previous one. Firstly, the PIV Tool loads an image sequence, lets the user select one or more square regions of interest (ROIs) around the cells (350 pixels by default), and runs the PIV algorithm to compute the velocity field inside each ROI, frame by frame. Secondly, the Process Tool takes the PIV results and computes, for each ROI and each frame, the mean velocity magnitude in the central part of the cell, defined as the central 50% of the ROI in each direction, saving it as a time series together with a plot. Finally, the Viewer Tool is used to explore the results: it overlays the vector field on the original frames, computes a set of derived quantities, and exports images, videos and data files.

Two corrections are available to deal with typical microscopy artifacts. Unsteady illumination can be compensated with a de-flickering option, and the row misalignment typical of scanning laser confocal microscopes can be estimated and corrected automatically for each ROI before the PIV is run.

3.4.2 The PIV algorithm

Particle Image Velocimetry was developed in experimental fluid dynamics [64]. In its classic form, a flow is seeded with small tracer particles, a plane of the fluid is illuminated with a laser, and two images are taken a short time apart; comparing them gives the velocity field of the flow. However, the algorithm itself, is not specific to particles: it can estimate the displacement between any two greyscale images that contain enough texture. In the cytoplasm, the role of the tracer particles is played by organelles and granules, which give the image a pattern that moves with the flow.

The first image of each pair is divided into a grid of small square sub-windows, called interrogation regions. For each region, the algorithm looks for the displacement $(\Delta x, \Delta y)$ that best matches its pixel pattern in the second image, by computing the cross-correlation between the two:

$$R(\Delta x, \Delta y) = \sum_i \sum_j I_1(i, j) I_2(i + \Delta x, j + \Delta y), \quad (3.1)$$

where I_1 and I_2 are the pixel intensities of the region in the two frames. The position of the highest peak of R gives the most likely displacement of the region, and dividing it by the time between the frames gives a velocity vector. Repeating this for every interrogation region produces a full vector field (Figure 3.1).

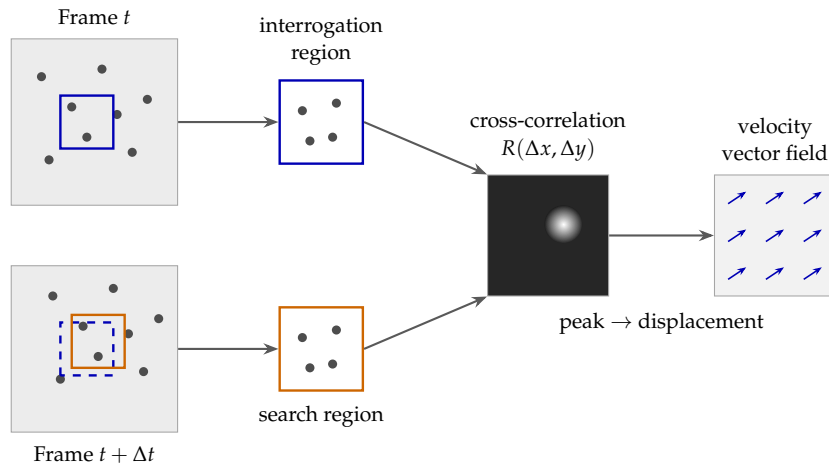


Figure 3.1: Principle of Particle Image Velocimetry as implemented in CellPIV. Each interrogation region of frame t is cross-correlated with the corresponding area of frame $t + \Delta t$; the position of the correlation peak gives the local displacement, and repeating the procedure over all regions yields the velocity field. Based on the scheme in [2].

CellPIV relies on MatPIV, an open-source MATLAB implementation of PIV [65], using its multi-pass cross-correlation scheme with a 50% overlap between neighboring regions. In the multi-pass scheme, the interrogation regions start large and are halved at each pass (the available sizes are 128, 64, 32, 16 and 8 pixels), with the displacement found at one pass used as the starting point of the search at the next. This combines the best of both worlds: large initial regions allow large displacements to be captured, while small final regions give a fine spatial resolution. The 50% overlap doubles the density of the vector field. However, Smaller regions contain fewer features, so the correlation peak becomes less distinct and the signal-to-noise ratio decreases, producing more erroneous vectors.

3.4.3 Derived metrics

From the vector field of each frame, CellPIV computes a set of scalar quantities that summarize the motion of the cytoplasm over time [2]. Plotted frame by frame, each

of them becomes a time series:

- **Mean magnitude:** the mean length of the velocity vectors in the frame, in pixels per frame. It measures how fast the cytoplasm is moving overall.
- **Standard deviation of direction:** for each vector, the inverse of the standard deviation of the vector angles in its 3×3 neighborhood, averaged over the frame. High values mean that groups of vectors point in the same direction (an ordered, coherent flow), low values that the vectors are randomly oriented.
- **Vorticity:** the mean local rotation of the velocity field, which highlights areas of rotational flow; positive values indicate clockwise rotation, negative values anticlockwise rotation.
- **Hybrid:** a combination of mean magnitude and inverse standard deviation of direction, which is high when the cytoplasm moves fast and/or coherently.
- **Sum mean magnitude:** the mean magnitude of the vectors summed over a given number of frames forward in time, which captures the net displacement accumulated over a short window rather than the instantaneous speed.

Besides these summary metrics, the Viewer Tool can draw streamlines and path lines (integrated with Runge-Kutta schemes) to visualize the trajectories that the cytoplasm would follow, and it can import calcium imaging data to compare them with the motion signal.

These are exactly the kinds of metrics used in this thesis. As described in Section 6.2, used time series are summary statistics of the motion field of each frame, computed both with the PIV approach of CellPIV and with two differential optical flow methods.

3.4.4 Strengths and limitations

CellPIV is non-invasive, since it only needs the images that the incubator already records. It is grounded in a clear biological mechanism, the calcium-driven contractions of the cytoskeleton. And it can provide information very early, potentially within hours of fertilization, when no morphological criterion is yet available. Its metrics, being physical quantities, are also directly interpretable.

It also has limitations. The original study was performed on mouse zygotes imaged every 10 seconds, whereas clinical time-lapse incubators usually acquire human embryos every 10–20 minutes. The software requires manual selection of the ROIs and careful parameter choices, PIV processing can take hours for long videos, and the predictive model in the original study was a simple statistical association rather than a trained classifier. These limitations motivate the approach of this thesis: extracting motion series automatically from routine time-lapse videos over the whole culture period, and feeding them to modern time series classifiers trained on thousands of embryos.

3.5 Comparison and limitations of these systems

Looking at these systems together, a few limitations can be observed. Firstly, commercial models, owe much of their performance to very large multi-center datasets, which only a company can assemble through commercial agreements with many clinics. Furthermore, those systems are closed: they cannot be inspected, retrained or adapted, and their explanations, when present, are limited to what the manufacturer chooses to expose. Research frameworks, on the other hand, are usually trained and validated on data from one or a few centers, so whether they work as well in other laboratories, with different incubators, protocols and patients, remains an open question. Collecting multi-center data in one place runs into the regulatory obstacles discussed in Chapter 4.

A second limitation concerns generalizability, i.e. whether a model keeps its performances in clinics other than those in which it has been developed. For iDAScore as shown before, different studies has been made resulting in no evidence on collapse in performance.

For the other systems this evidence is missing; in fact, BlastAssist, EmbryoMLSelection and CellPIV has been developed and tested on a single center with no external test set [20, 21, 45]. It's important to underline that does not means that these systems generalize poorly, but it simply has not been tested.

This thesis tries to address these points. It builds on the CellPIV idea of describing embryo development through motion signals, extracts them automatically from routine time-lapse videos, and trains open time series classifiers on them. It does so with Federated Learning, so that several centers could contribute to the same model without sharing their data, and pairs the federated model with an explainability procedure designed for this setting.

Table 3.1: Summary comparison of the decision-support systems discussed in this chapter.

	iDAScore	CHLOE	BlastAssist	EmbryoML- Selection	CellPIV
Type	Commercial	Commercial	Research	Research	Research
Main input	Time-lapse sequences	Time-lapse videos (3 focal planes)	Static images (day 5/6)	Morphokinetic parameters	High-frame-rate zygote videos
Core method	Deep learning on image sequences	Segmentation (Mask R-CNN) + morphometry	U-Net + CNN	ML on morphokinetic features	PIV (cross-correlation)
Output	Implantation score	Blastulation, implantation, ploidy estimates	Segmentation maps, Gardner-like score	Blastocyst development	Cytoplasmic motion metrics
Timing	Up to blastocyst	Up to blastocyst	Late (blastocyst)	Early to intermediate	Very early (hours after fertilisation)
Open method	No	Partly published	Yes	Yes	Yes

Chapter 4

Federated Learning

Federated Learning (FL) [29] is one of the most recent advancements of distributed machine learning in recent years. It offers a methodological answer to a very practical problem: how to learn from data that cannot be centralized. In conventional machine learning, the raw data have to be copied to a single machine or a central repository before training can start. This creates bottlenecks, and in many domains it clashes with legal and ethical constraints.

FL turns this around. The central server no longer collects data. It sends the model to the participating nodes, usually called clients, and each client trains it on its own data. What travels back to the server are only parameter updates, such as weights or gradients. By repeating this cycle of local training, aggregation on the server and redistribution of the new parameters, the federation converges to a global model that generalizes across clients, while the data never leave their original location [30].

Table 4.1 shows the main difference between the classic approach and the federated one.

4.1 Problem formulation

Consider K clients, each holding a local dataset $\mathcal{D}_k = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_k}$ with n_k samples, and let $n = \sum_{k=1}^K n_k$. For a model with parameters $\mathbf{w} \in \mathbb{R}^d$ and a loss $\ell(\mathbf{w}; \mathbf{x}, y)$, the local empirical risk of client k is

$$F_k(\mathbf{w}) = \frac{1}{n_k} \sum_{(\mathbf{x}, y) \in \mathcal{D}_k} \ell(\mathbf{w}; \mathbf{x}, y). \quad (4.1)$$

FL aims to minimize the global risk, defined as the weighted average of the local ones:

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) = \sum_{k=1}^K \frac{n_k}{n} F_k(\mathbf{w}). \quad (4.2)$$

If all the data have been pooled, Equation (4.2) would simply be the empirical risk of centralized training. What makes FL different is that the server can neither

evaluate F_k directly nor see $\mathcal{D}_k$; it can only ask clients to run computations locally and report the results.

4.2 Aggregation strategies

How well and how stably a federation converges depends to a large extent on how the server aggregates the client updates. Many strategies have been proposed, each with its own strengths and weaknesses.

FedAvg. Better explained in Section 4.3, FedAvg is the most common choice [29]: the server takes a weighted average of the client weights, proportional to the size of each local dataset. It is cheap in terms of communication and computation and works well when clients are similar, but it struggles when the data are strongly heterogeneous.

FedProx. Federated Proximal (FedProx) [66, 67] was introduced to address that weakness. It adds a proximal term to the local loss that keeps each client model from drifting too far from the global one:

$$\min_w F_k(w) + \frac{\mu}{2} \|w - w^t\|^2. \quad (4.3)$$

This makes FedProx considerably more robust to statistical heterogeneity and to clients with different computing power or unreliable connections. However, it has an extra hyperparameter, μ , that has to be tuned carefully.

Personalized FL. Personalized Federated Learning (PFL) [68] covers a family of hybrid strategies that combine a shared global structure with local adaptations, trying to balance collective generalization and accuracy on each individual client.

FedOpt and beyond. FedOpt [69, 70] brings adaptive optimizers such as Adam and Adagrad to the server side to speed up learning, and other approaches rely on clustering clients or on knowledge distillation. Which method works best depends on the network, the infrastructure, the available compute and, above all, on how heterogeneous the data are.

In this work FedAvg has been used for every model. It is the reference algorithm in the literature and has almost no extra hyperparameters, which makes it a clean baseline for measuring what we lose, if anything, by not centralizing the data.

4.3 Federated Averaging

The most widely used algorithm for implementing Equation (4.2) is Federated Averaging (FedAvg) [29]. At each round t , the server sends the current global model w^t to a subset S_t of clients. Each of them starts from w^t , runs E epochs of stochastic gradient descent on its own data and obtains w_k^{t+1} . The server then averages the models it receives, weighting each by its number of samples:

$$w^{t+1} = \sum_{k \in S_t} \frac{n_k}{\sum_{j \in S_t} n_j} w_k^{t+1}. \quad (4.4)$$

The full procedure is given in Algorithm 1. Compared with other strategies as FedOpt, where clients communicate after every step, FedAvg saves a great deal of communication because each client does several epochs of local work between two synchronizations.

Algorithm 1: Federated Averaging (FedAvg)

Input: number of rounds T , local epochs E , learning rate η , client fraction C

Output: global model w^T

```

1 Server initializes  $w^0$ ;
2 for  $t = 0, \dots, T - 1$  do
3   Server selects a set  $S_t$  of  $\max(\lceil C \cdot K \rceil, 1)$  clients;
4   foreach client  $k \in S_t$  in parallel do
5      $w_k \leftarrow w^t$ ;
6     for  $e = 1, \dots, E$  do
7       foreach mini-batch  $b \subset \mathcal{D}_k$  do
8          $w_k \leftarrow w_k - \eta \nabla \ell(w_k; b)$ ;
9       Send  $(w_k, n_k)$  to the server;
10   $w^{t+1} \leftarrow \sum_{k \in S_t} \frac{n_k}{\sum_{j \in S_t} n_j} w_k$ ;
```

4.4 Heterogeneous (non-IID) data

Centralized training usually assumes that samples are independent and identically distributed (IID). In FL this assumption almost never holds, because each client sees its own distribution $P_k(x, y)$, shaped by its patients, equipment and practices [30, 71]. The literature usually distinguishes four kinds of dataset shifts. With *label skew* (or prior shift), the class proportions $P_k(y)$ differ between clients. With *feature skew* (covariate shift), the input distribution $P_k(x)$ changes, for example because of different incubators or acquisition settings. With *concept shift*, the relation $P_k(y | x)$ itself differs, for instance because annotation criteria are not the same everywhere. Finally, with *quantity skew*, clients simply hold very different amounts of data.

Such statistical heterogeneity makes local updates drift apart, a phenomenon known as client drift. Each local model is pulled towards its own optimum $\arg \min F_k$, which may be far from the global one, slowing down the aggregation strategy convergence and lowering its final performance [71, 72].

4.5 Privacy

In medicine, the way FL is built helps with two long-standing problems: patient privacy and the fragmentation of health data [31, 73]. Centralized training requires the AI developer to follow strict data protection rules, namely the GDPR [25, 26], the MDR/IVDR regulations on medical devices [27] and the AI Act [28], which severely limit how health-related data can leave the hospital that collected them.

FL protects privacy at the source: sensitive clinical information never leaves the IT infrastructure of each client. However, sharing only model parameters, is not a complete guarantee in itself: under some conditions, training data can be partly reconstructed from shared gradients [74]. To reduce the risk of reconstruction or re-identification attacks, FL is commonly combined with Differential Privacy [75], which adds calibrated noise to the updates and formally bounds what can be learned about any single sample, and with secure aggregation [76] and encryption; it lets the server compute the sum of the updates without seeing any individual contribution.

Beyond privacy, FL also changes how clinical research can be organized, because it makes large multi-institutional and even cross-border collaborations much easier [32]. Models trained on large, diverse patient cohorts from different regions are less prone to overfitting and to the selection bias typical of single-center studies. It is also lighter on infrastructure: sending parameter vectors takes far less bandwidth than continuously moving high-resolution diagnostic files around.

Table 4.1: Main differences between centralized and federated learning.

	Centralized learning	Federated learning
Privacy	Requires complex data transfer agreements	Health data never leave the hospital
Dataset size	Limited to the centers willing to share raw data	Collaboration among many centers, also across borders
Model bias	Risk of overfitting homogeneous local populations	Reduced, thanks to heterogeneous and geographically diverse cohorts
Bandwidth	High, to move large images and videos	Low: only model parameters are exchanged
Complexity	Simple infrastructure, a single compute node	Distributed orchestration, heterogeneous and non-IID clients

4.6 The Flower framework

As better outlined in Chapter 8, the federated learning environment developed in this thesis has been developed with Flower(`flwr`) [77], an open-source FL framework that does not depend on any particular deep learning library. Flower clearly separates two components. The **ServerApp** defines the aggregation strategy (FedAvg in our case), the number of rounds and, optionally, a function to evaluate the global model on the server side. The **ClientApp** contains what runs on each node: loading the local data, training from the parameters received, and local evaluation. Server and clients exchange messages that carry model parameters (`ArrayRecord`), metrics (`MetricRecord`) and configuration values such as the learning rate (`ConfigRecord`).

Flower also features a Ray-based simulation engine capable of running multiple virtual clients on a single machine. This specific mode was utilized for all experiments in this study. A major advantage of this framework is that the same codebase can be seamlessly transitioned to a real distributed deployment without altering the underlying training logic.

Chapter 5

Explainable AI in a federated setting

Federated Learning gives ART a solid basis for multi-center collaboration: clinics can train predictive models together without handing over their data. However, the deep learning models involved, are black-boxes. When using AI to support clinical decisions, such as which embryo to implant, it is not enough to know how likely it is to reach the blastocyst stage; the clinician also needs to know which developmental dynamics led the model to that conclusion. This chapter introduces the main ideas of Explainable AI (XAI) and then discusses what changes when the model to be explained is federated. On this basis SHAP-FL, the procedure used in this thesis is defined.

5.1 Why explanations matter

There are at least four reasons why explainability matters in medicine.

Firstly, trust: an embryologist is unlikely to follow a bare number, especially when it contradicts their own reading of the embryo's morphology. If the system can show which phases of development drove its prediction, the clinician can check that reasoning against their own knowledge and decide, knowingly, whether to accept it [13, 33].

Secondly, accountability and regulation: under European law, AI systems used as medical devices, or as parts of them, are considered high-risk and come with transparency obligations: users must be able to interpret the output and use it appropriately [27, 28]. The GDPR also contains specific safeguards for decisions based on automated processing [25].

Thirdly, debugging and bias detection: explanations help developers spot models that lean on spurious correlations, such as acquisition artifacts, administrative variables, or information that would not be available at decision time, instead of biologically sensible signals. In a multi-center setting this check is even more important, since each center can bring its own artifacts.

Finally, scientific insight: looking at what an accurate model has learned can suggest new biological hypotheses, for example by highlighting time windows that

are especially informative about an embryo’s fate.

5.2 Explanation methods

XAI methods are usually classified along a few dimensions [34, 78].

Intrinsic or post-hoc. Some models are interpretable by design: linear models, shallow decision trees, additive models. Post-hoc methods, on the other hand, explain a model after it has been trained, treating it fully or partly as a black-box. High-stakes applications has been argued to prefer intrinsically interpretable models [79]. However, when the input is complex, as with a time series of several hundred samples, deep models are often much more accurate, and post-hoc explanations become a pragmatic compromise.

Local or global. A local explanation justifies one prediction (“why was this embryo rated as promising?”), while a global explanation describes the model’s overall behavior (“which features matter most in general?”). Global explanations can be built by aggregating many local ones.

Model-agnostic or model-specific. Agnostic methods such as LIME [80] and KernelSHAP [35] only need to query the model. Specific methods exploit its internals, for example the gradients of a neural network [81, 82], and are usually more efficient.

Form of the explanation. Explanations can take the form of feature attributions, representative examples, counterfactuals (“what is the smallest change to the input that would flip the prediction?”), or rules.

In this thesis post-hoc methods based on Shapley values have been used, which assign each input feature a numerical contribution to the prediction. These methods have been chosen for their sound theoretical properties and because attributions are additive, which, as we will see, makes them easy to aggregate in a federated setting.

5.3 Shapley values

Shapley values come from cooperative game theory [83]. Given a set of players $N = \{1, \dots, M\}$ and a characteristic function $v : 2^N \rightarrow \mathbb{R}$ that gives the payoff of every coalition $S \subseteq N$, with $v(\emptyset) = 0$, the question is how to share the total payoff $v(N)$ fairly among the players. The Shapley value of player i is

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|! (M - |S| - 1)!}{M!} [v(S \cup \{i\}) - v(S)]. \quad (5.1)$$

The term in square brackets is the marginal contribution of i to coalition S , and the combinatorial weight is the probability that, in a random ordering of the players, exactly the members of S come before i . In conclusion, ϕ_i is the average marginal contribution of i over all possible orders in which the coalition could be formed.

5.4 SHAP: SHapley Additive exPlanations

SHAP [35] implements this idea over to the explanation of model predictions: the input features become the players and the model's output becomes the payoff to be shared.

5.4.1 Additive feature attribution

Let $f : \mathbb{R}^M \rightarrow \mathbb{R}$ be the model to explain (for a classifier, typically the logit or the probability of one class) and $\mathbf{x}$ the instance of interest. SHAP belongs to the family of additive feature attribution methods, which explain $f(\mathbf{x})$ through a linear model over simplified binary variables $\mathbf{z}' \in \{0, 1\}^M$ indicating whether each feature is present or absent:

$$g(\mathbf{z}') = \phi_0 + \sum_{j=1}^M \phi_j z'_j. \quad (5.2)$$

Lundberg and Lee [35] showed that there is exactly one additive attribution with three desirable properties: firstly, local accuracy: meaning that the explanation reproduces the prediction exactly, $f(\mathbf{x}) = \phi_0 + \sum_j \phi_j$; secondly, missingness, meaning that absent features get zero attribution; finally, consistency, meaning that if a model changes so that a feature's marginal contribution grows or stays the same in every coalition, that feature's attribution does not decrease. This attribution is given by the Shapley values of the game whose characteristic function is

$$v_{\mathbf{x}}(S) = \mathbb{E}[f(\mathbf{X}) \mid \mathbf{X}_S = \mathbf{x}_S], \quad (5.3)$$

that is, the expected prediction when only the features in S are known. The base value $\phi_0 = \mathbb{E}[f(\mathbf{X})]$ is the average prediction over the reference data.

5.4.2 The background dataset

Computing the expectation in Equation (5.3) requires removing the features that are not in S . In practice they are replaced by values taken from a background dataset $\mathcal{B}$ that represents the data distribution:

$$v_{\mathbf{x}}(S) \approx \frac{1}{|\mathcal{B}|} \sum_{\mathbf{b} \in \mathcal{B}} f(\mathbf{x}_S, \mathbf{b}_{\bar{S}}). \quad (5.4)$$

The choice of background defines the question the explanation answers: SHAP values explain how far the prediction for $\mathbf{x}$ is from the average prediction over the

background. This detail turns out to be crucial in FL, because the background is made of real data, and in a federation real data have to stay local.

5.4.3 Estimating SHAP values

Computing Equation (5.1) exactly means evaluating 2^M coalitions, which is out of reach even for a few dozen features, so several estimators have been developed. **KernelSHAP** [35] is model-agnostic and estimates Shapley values through a weighted linear regression over sampled coalitions; however, it is expensive. **TreeSHAP** [84] computes exact values in polynomial time for tree-based models. **DeepSHAP** [35], available as DeepExplainer, is designed for neural networks and builds on DeepLIFT [81]. DeepLIFT propagates differences in activation with respect to a reference input back through the network, using layer-wise multipliers, and its attributions satisfy summation-to-delta: $\sum_j \phi_j = f(\mathbf{x}) - f(\mathbf{b})$. DeepSHAP reads DeepLIFT’s propagation rules as a compositional approximation of Shapley values and averages the attributions over every sample in the background. Its cost is comparable to a few backward passes per sample, so it scales well to high-dimensional inputs.

The model explained in this work is a neural network (Section 7.4.3), so DeepSHAP has been used.

5.4.4 From local to global explanations

SHAP values are local: each instance x_i gets its own vector $\phi_i \in \mathbb{R}^M$. A global importance score for feature j is obtained by averaging the absolute values over a set of instances $\mathcal{X}$:

$$I_j = \frac{1}{|\mathcal{X}|} \sum_{x_i \in \mathcal{X}} |\phi_{ij}|. \quad (5.5)$$

Absolute values are taken so that contributions of opposite sign on different instances do not cancel out. A feature that pushes some predictions strongly towards the positive class and others strongly towards the negative one, is still very important to the model.

5.5 Explainability in Federated Learning

Applying post-hoc explanation methods to a federated model raises problems that simply do not exist in the centralized case. Everyone knows the global model f_w , but the data needed to explain it, both the instances and the background, are spread across the clients and cannot be pooled.

5.5.1 Challenges

The first issue is the **local background**. Since SHAP values depend on the reference data, computing them on the server would require sending real data there, which defeats the purpose of FL. A synthetic or public background, on the other hand, may not represent the real distribution well.

The second is that **explanations can differ between clients**. With non-IID data, the same global model may rely on different features in different centers, because their embryos and acquisition conditions are not the same. A single global explanation risks hiding these differences, which are actually useful: they can point to a center with artifacts, or to a subgroup of patients where the model behaves differently.

The third is that **explanations can leak information too**. They are derived from data, and per-instance SHAP values, if shared, may reveal something about the original samples. Some level of aggregation is needed before anything is sent.

5.6 How SHAP-FL works

Let f_{w^T} be the global model obtained at the end of federated training. SHAP-FL [85] works on two levels, as shown in Figure 5.1.

At the **local level**, each client k already holds the global model f_{w^T} from the last round. It draws a local background $\mathcal{B}_k$ by sampling instances from its training set, and a set of instances to explain, $\mathcal{X}_k$, from its test set. For each $x_i \in \mathcal{X}_k$ it computes with DeepSHAP the attribution vector $\phi_i^{(k)} \in \mathbb{R}^M$ for the positive class (blastocyst formation), and from these it computes the local importance of every feature:

$$I_j^{(k)} = \frac{1}{|\mathcal{X}_k|} \sum_{x_i \in \mathcal{X}_k} |\phi_{ij}^{(k)}|, \quad j = 1, \dots, M. \quad (5.6)$$

Only the vector $I^{(k)} \in \mathbb{R}^M$ and a weight m_k , such as the number of explained instances $|\mathcal{X}_k|$ or, as in FedAvg, the size of the local training set, are sent to the server. The per-instance explanations stay with the client, where the local embryologist can look at them as explanations of individual clinical cases.

At the **global level**, the server combines the vectors it receives with a weighted average, just as FedAvg does for model parameters:

$$\bar{I}_j = \sum_{k=1}^K \frac{m_k}{\sum_{l=1}^K m_l} I_j^{(k)}. \quad (5.7)$$

If all clients used the same background, $\bar{I}_j$ would be exactly the importance of Equation (5.5) computed over the union of the sets $\mathcal{X}_k$, as if the data had been centralized. In the general case, with local backgrounds, $\bar{I}_j$ should be read as the average importance of feature j relative to the reference distribution of each center. This is actually the right question from a clinical point of view: how much does this

feature matter in explaining a prediction, compared with a typical embryo from the same laboratory?

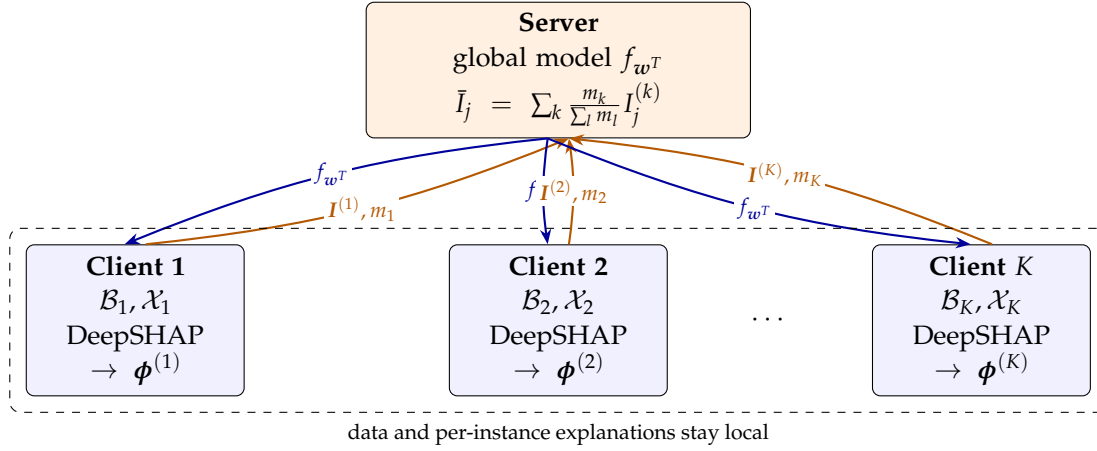


Figure 5.1: The SHAP-FL procedure. Each client explains the global model on its own data with a local background and shares only its aggregated importance vector; the server computes the global importance as a weighted average. Image generated with Claude

5.6.1 Properties

SHAP-FL has a few properties worth pointing out. In terms of **privacy**, the only thing communicated is a vector of M averages of absolute values, each computed over tens or hundreds of instances; neither raw data nor per-instance attributions leave the client. If needed, $I^{(k)}$ can be protected with secure aggregation or perturbed with differentially private noise before sending (Section 4.5). In terms of **cost**, the procedure needs just one extra communication round at the end of training, and running DeepSHAP on a client costs about as much as one epoch of inference. It also gives **two levels of reading** at the same time: local explanations, per embryo and per center, and a global explanation of the whole federation. Finally, comparing the vectors $I^{(k)}$ across clients lets us **measure heterogeneity**, that is, how similarly the global model reasons in different centers.

5.6.2 Measuring agreement between explanations

To quantify how consistent the explanations are across centers we use three complementary measures. The **Spearman rank correlation** $\rho(I^{(k)}, I^{(l)})$ between the importance vectors of two clients measures agreement on the overall ranking of the features. The **Jaccard index** between the sets $\mathcal{T}_q^{(k)}$ and $\mathcal{T}_q^{(l)}$ of the q most important features,

$$J_q(k, l) = \frac{|\mathcal{T}_q^{(k)} \cap \mathcal{T}_q^{(l)}|}{|\mathcal{T}_q^{(k)} \cup \mathcal{T}_q^{(l)}|}, \quad (5.8)$$

measures agreement on the top features, the ones a clinician would typically look at. The **concentration curve** of the global importance, i.e. the cumulative share of $\sum_j \bar{I}_j$ covered by the first q features in order of importance, tells us if the model relies on a few dominant features or on information spread across many of them.

5.7 Explanation level

One last point concerns what the explanation refers to. When SHAP is applied directly to the input of a time series classifier, the features are the time points of the optical-flow series. The attribution is then easy to read in biological terms, because it tells us which windows of embryo development influenced the prediction.

When the classifier works on a latent representation produced by a foundation model, as with the 1024-dimensional MOMENT embeddings used in this work (Section 7.4.3), SHAP attributes importance to the embedding dimensions, which have no direct meaning. The explanation is still useful for studying how the model is structured, how much centers agree and how concentrated the information is, but to become clinically meaningful it has to be projected back onto the original time axis. One way to do this is to propagate attributions through the encoder; another is to compute attributions by occluding time windows of the input series. This choice and its implications are explained in Chapter 9, and again among the future directions in Chapter 10.

Chapter 6

Dataset and data preparation

This chapter describes the data used: where they come from, which variables are available, how optical-flow time series are obtained from time-lapse videos, which quality checks were applied, and how the data were partitioned to simulate a federated environment.

6.1 Origin and content

The dataset comes from the Laboratory of Reproductive Biology and Biotechnology of the University of Pavia, in collaboration with the Genera assisted reproduction clinic in Rome. It contains 5011 embryos cultured in time-lapse incubators and recorded between 2013 and 2020. For each embryo we have the optical-flow time series extracted from its time-lapse video (Section 6.2); clinical variables such as maternal age, semen quality and culture medium; the morphokinetic annotations made by the embryologists, namely the number of pronuclei (PN), the times of pronuclear appearance and fading (t_{PNa} , t_{PNf}), the division times ($t_2, \dots, t_{9+}$), the time of morula compaction (t_M) and the times of blastocyst start, formation and expansion (t_{SB} , t_B , t_{EB}); laboratory identifiers for the dish, the well and the combined dish_well code, which includes the acquisition date; and, for biopsied embryos, the result of the preimplantation genetic test (euploid or aneuploid).

The target is the binary variable BLASTO_{NY}, which tells whether the embryo reached the blastocyst stage ($y = 1$) or not ($y = 0$). Overall, 50.6% of the embryos form a blastocyst, so the problem is essentially balanced globally.

Cleaning. Some cleaning and standardization were needed before the data could be used. An issue was patients with more dishes, which is common in cycles where many oocytes are retrieved, since a dish holds at most 12 wells. All wells belonging to the same patient and cycle were given the same identifier (patient_id), so that embryos from one patient could be kept together during evaluation.

6.2 Optical-flow time series

To quantify cytoplasmic motion, we apply optical flow algorithms to the frames of the time-lapse videos. Optical flow estimates, for each pixel or region, the apparent displacement (u, v) between two consecutive frames, under the assumption that brightness is preserved:

$$I(x, y, t) = I(x + u, y + v, t + \Delta t) \implies I_x u + I_y v + I_t \approx 0, \quad (6.1)$$

where I_x , I_y and I_t are the partial derivatives of the image intensity. Equation (6.1) gives a single constraint for two unknowns (the so-called aperture problem), and the various methods differ in the extra assumptions they make to solve it.

The extraction step has been executed with two different methods. The **Farneback method** [86] is dense and multi-scale: it approximates the intensity around each pixel with a quadratic polynomial and uses a Gaussian image pyramid to capture motion at different scales while avoiding aliasing. The **Lucas-Kanade method** [87] is local: it assumes the flow is constant within a small patch and solves the resulting over-determined system by least squares,

$$\begin{bmatrix} u \\ v \end{bmatrix} = (A^\top A)^{-1} A^\top \mathbf{b}, \quad A = \begin{bmatrix} I_x(\mathbf{p}_1) & I_y(\mathbf{p}_1) \\ \vdots & \vdots \\ I_x(\mathbf{p}_n) & I_y(\mathbf{p}_n) \end{bmatrix}, \quad \mathbf{b} = - \begin{bmatrix} I_t(\mathbf{p}_1) \\ \vdots \\ I_t(\mathbf{p}_n) \end{bmatrix}. \quad (6.2)$$

From the velocity vectors computed frame by frame with each method, we derive a few summary metrics in the form of time series. Firstly, the Mean Magnitude: the average speed of the field, $\frac{1}{|\Omega|} \sum_{\mathbf{p} \in \Omega} \sqrt{u^2 + v^2}$ over the region of interest Ω . Secondly, Vorticity: $\omega = \partial v / \partial x - \partial u / \partial y$, measures local rotation and reveals vortices. Thirdly, Standard Deviation: captures how dispersed the magnitudes are, and hence how uniform the motion is. Furthermore, Hybrid: combining magnitude and dispersion. Finally, Sum Mean Magnitude: is a moving average of motion intensity over consecutive frames.

Figure 6.1 shows the median series for the two classes. In the first hours the two groups are almost indistinguishable. Between roughly 20 and 40 hours we see peaks related to the first mitotic divisions, more pronounced in embryos that go on to form a blastocyst. From about 90 hours, as compaction and cavitation begin, the two classes separate clearly: embryos that become blastocysts show a marked increase in activity, which is what one would expect as the blastocoel expands.

6.3 Data quality and preprocessing

Careful preprocessing is essential to prevent visual noise or gaps in time from spoiling training. Before the flow is extracted, the frames are contrast-enhanced to even out the lighting, so that the embryo's outline can be traced precisely and the region of interest (ROI) in which vectors are computed can be isolated. The videos

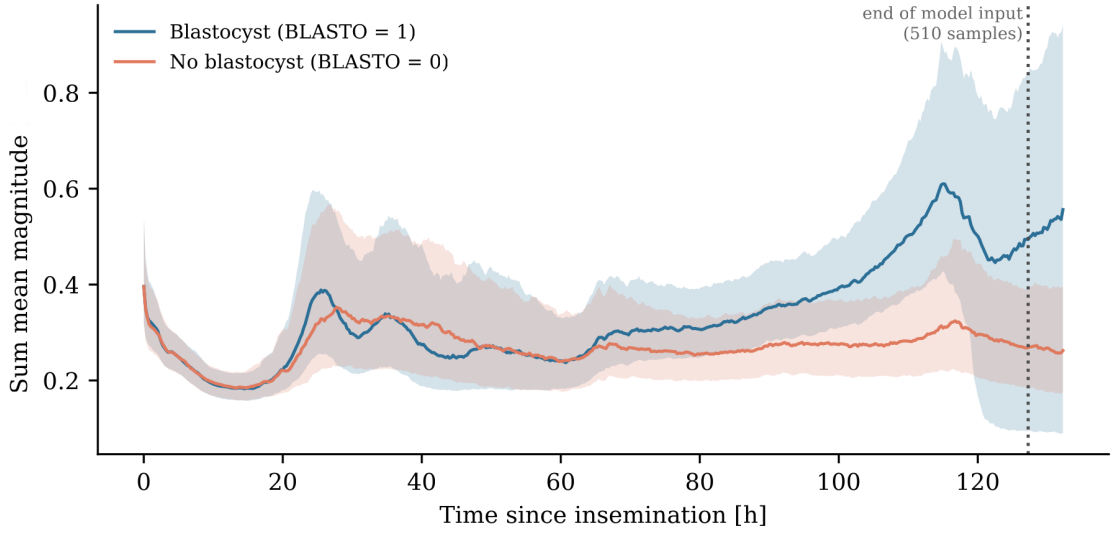


Figure 6.1: Median (line) and interquartile range (shaded area) of the optical-flow series for embryos that do (blue) or do not (orange) reach the blastocyst stage, computed on the training partitions. The dotted line marks the last time point fed to the models.

are then screened, and damaged, out-of-focus or artifact-ridden sequences that would distort the spatio-temporal derivatives of the flow are discarded. Because incubator acquisitions can be paused and recordings vary in length, the resulting series are aligned and resampled onto a common 15-minute grid. A normalization step follows (for example quantile normalization), which serves both to make the sequences uniform for the different ML architectures and to damp anomalous peaks or amplitude changes that do not reflect real cytoplasmic behavior.

Model input. All neural architectures receive the same input vector, built from the numeric columns of the dataset and truncated to at most 512 values, which is MOMENT’s native context length (Section 7.4). In the current implementation this vector contains, in order, the numeric patient identifier, the maternal age and the first 510 samples of the optical-flow series (from 0 h to 127.25 h).

6.4 Simulating a federated environment

6.4.1 How many partitions? A bootstrap analysis

Before settling on the final partitioning, how model performance degrades as the number of centers grows has been checked. This operation has been done with a bootstrap procedure. The dataset was first split into a training set (80%) and a test set (20%), stratified by class. Then, for each $k = 1, \dots, 15$, the training set was divided into k stratified partitions, and from each partition, a sample of

$\lfloor n_{\text{train}}/k \rfloor$ elements keeping the class proportions has been extracted. On each of these a reference classifier (a 100-tree Random Forest preceded by imputation and standardization) has been trained and evaluated on the common test set. Averaged accuracy and F1-score over the k partitions have been computed.

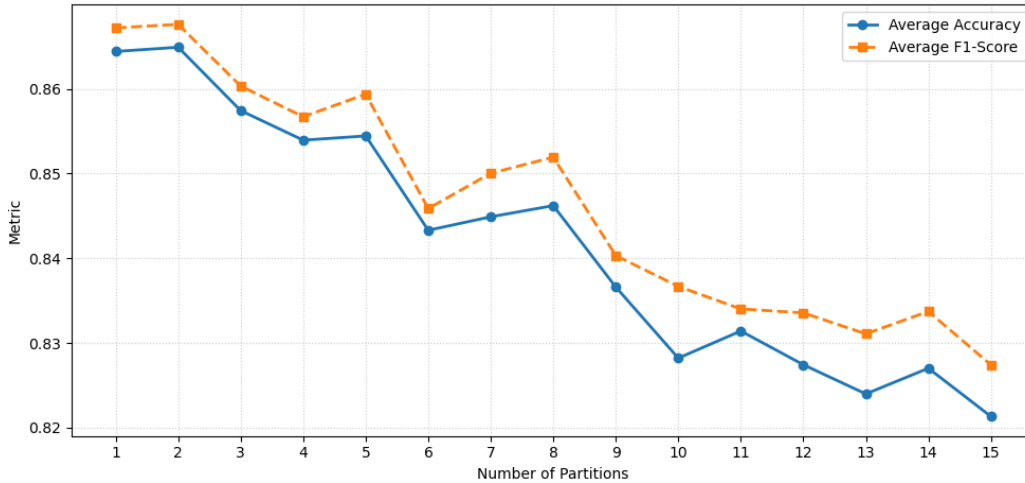


Figure 6.2: Average accuracy and F1-score as a function of the number of partitions, from the bootstrap analysis. More partitions mean fewer samples per node; beyond 8 partitions the metrics drop markedly.

As Figure 6.2 shows, the metrics decrease slowly up to 8 partitions and then fall more sharply, dropping below 0.84 accuracy for $k \geq 9$. With about 5000 embryos, then, it does not make sense to spread the data across more than 7–8 centers, or each center will not have enough data. Our year-based split, with 7 clients plus a test year, stays within this limit.

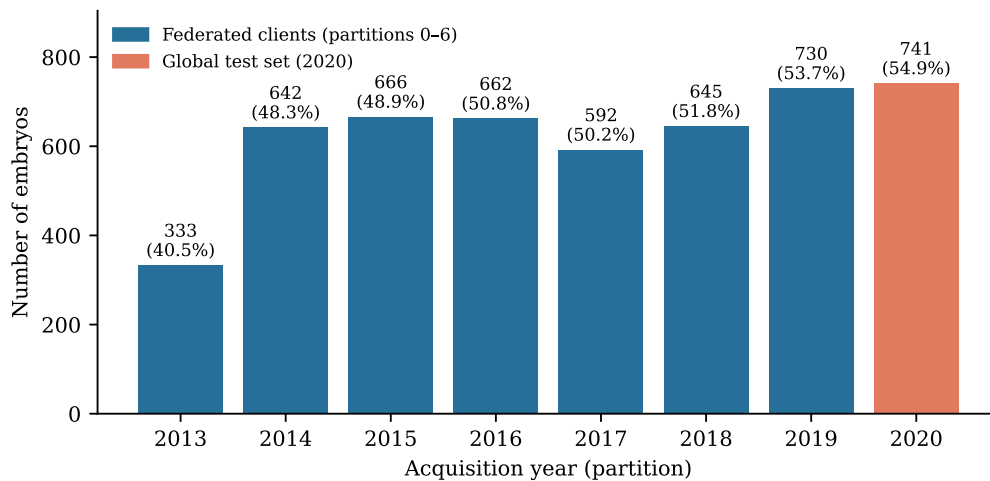
6.4.2 Partitioning

The years 2013 to 2019 make up the seven federated clients (partitions 0–6), while the 2020 embryos form the global test set and are used only for the final evaluation. No patient identifier in the test set appears in any training partition. Table 6.1 and Figure 6.3 summarize the partitions.

Two of the forms of heterogeneity discussed in Section 4.4 are clearly visible in Table 6.1. There is **quantity skew**, since the 2013 partition has about half as many embryos as the others. And there is **label skew** with a time trend: the share of blastocysts rises almost steadily from 40.5% in 2013 to 54.9% in 2020, probably thanks to gradual improvements in culture conditions. As a result, the 2020 test set has a higher proportion of positive cases than any training year. This prior shift makes the evaluation tougher, but also more realistic.

Table 6.1: Composition of the partitions. Partitions 0–6 are the federated clients; 2020 is the global test set.

Partition	Year	Embryos	Patients	Blastocysts (%)	Mean maternal age
0	2013	333	49	40.5	39.3
1	2014	642	103	48.3	39.6
2	2015	666	102	48.9	38.4
3	2016	662	120	50.8	38.7
4	2017	592	99	50.2	38.6
5	2018	645	98	51.8	38.1
6	2019	730	116	53.7	38.1
Test	2020	741	141	54.9	38.8
Total		5011	828	50.6	38.6

**Figure 6.3:** Number of embryos and percentage of blastocysts per partition.

Chapter 7

Time series classification models

The task we face is binary classification of univariate time series: given the optical-flow series $x = (x_1, \dots, x_L)$ of an embryo, the goal is to estimate the probability that the embryo reaches the blastocyst stage. The time series classification literature [88] offers very different families of methods. We picked four architectures that cover the main ones: ROCKET, which uses random convolutional kernels followed by a linear classifier; ConvTran, a hybrid of convolutions and Transformers; LSTM-FCN, a hybrid of recurrent and convolutional networks; and MOMENT, a large pre-trained foundation model, which we use both with full fine-tuning and as a frozen feature extractor.

These models differ in size, and in FL, this matters because the communication cost of each round is proportional to the number of parameters exchanged. The rest of the chapter describes each model and how it has been implemented.

7.1 ROCKET

ROCKET (RandOm Convolutional KErnel Transform) [89] rests on an idea that is as simple as it is effective. Instead of learning convolutional filters (as a CNN would) it generates a very large number of them *at random* and only learns a linear classifier on top of the resulting features.

Each kernel ω is defined by randomly drawn parameters: a length $l \in \{7, 9, 11\}$; weights $\omega_i \sim \mathcal{N}(0, 1)$, then centered to zero mean; a bias $b \sim \mathcal{U}(-1, 1)$; a dilation $d = \lfloor 2^a \rfloor$ with $a \sim \mathcal{U}(0, \log_2 \frac{L-1}{l-1})$, which allows patterns at different time scales to be picked up; and padding, applied or not with equal probability. Convolving the kernel with the series gives

$$z_t = b + \sum_{i=0}^{l-1} \omega_i x_{t+i \cdot d}, \quad (7.1)$$

from which two features are taken: the maximum, $\max_t z_t$, and the proportion of positive values, $\text{PPV} = \frac{1}{|z|} \sum_t \mathbb{1}[z_t > 0]$. With 10 000 kernels this yields 20 000 features per series.

In this implementation the ROCKET features are fed to a logistic regression. In the federated version every client applies the transform with the same random seed, so that all clients use identical kernels; this is what makes averaging the coefficients meaningful in the first place. Each client then fits the logistic regression on its own features, and FedAvg is applied only to the coefficient vector $\beta \in \mathbb{R}^{20000}$ and the intercept.

7.2 ConvTran

ConvTran [90] combines convolutional layers, which are good at capturing local patterns, with the self-attention of Transformers [91], which can model long-range dependencies. The original paper also proposes two position encodings designed for time series, where the order of the samples matters: time Absolute Position Encoding (tAPE) and efficient Relative Position Encoding (eRPE).

The version starts with a **convolutional embedding** made of two Conv1d layers (kernel length 9, 64 filters), each followed by batch normalization and a GELU activation. A **learnable positional encoding** is then added to the sequence, which goes through three **pre-norm Transformer encoder blocks**. Each block has Multi-Head Self-Attention (MHA, with 8 heads, dimension 64) and a Feed-Forward Network (FFN, with hidden size 256, GELU), each preceded by Layer Normalization (LN) and wrapped in a residual connections:

$$h' = h + \text{Dropout}(\text{MHA}(\text{LN}(h))), \quad (7.2)$$

$$h'' = h' + \text{FFN}(\text{LN}(h')), \quad (7.3)$$

where each attention head computes

$$\text{Attn}(Q, K, V) = \text{softmax}(QK^\top / \sqrt{d})V. \quad (7.4)$$

layer normalization rescales each time step independently, so that its features zero mean and unit variance. Normalizing the input of each sub-layer keeps the activations on a similar scale across blocks and keeps the training more stable. The **multi-head self-attention** lets every time step look at all the others and decide how much of them contribute to its new representation. Each output time step is weighted average of the values of all time steps. Using several heads allows the model to attend to different kinds of relations at the same time, for instance a short-range one and a long-range one. The **feed-forward network** is applied to each time step separately. It expands the representation to a hidden size of $d_{\text{ff}} = 256$, applies a GELU non-linearity and projects it back to dimension d .

The output is summarized by global average pooling and passed to an MLP classification head. It has been trained with RAdam and a learning rate of 10^{-5} .

7.3 LSTM-FCN

The original LSTM-FCN [92] runs two branches in parallel on the same series. One is a Fully Convolutional Network (FCN) [93], which extracts local, shift-invariant patterns; the other is an LSTM [94], which models temporal dependencies. The outputs of the two branches are concatenated and passed to a classifier.

We used a richer variant of this design, sketched in Figure 7.1. Each scalar sample x_t is first projected to 64 dimensions by a small two-layer MLP (Linear–LayerNorm–GELU–Dropout), then projected to 128 dimensions and normalized. A sinusoidal positional encoding [91],

$$PE_{(t,2i)} = \sin(t/10000^{2i/D}), \quad PE_{(t,2i+1)} = \cos(t/10000^{2i/D}), \quad (7.5)$$

is added to the sequence, followed by a 4-head self-attention layer with a residual connection and LayerNorm, so that each time step can draw on information from the whole series. The result feeds two branches. The **recurrent branch** is a two-layer bidirectional LSTM with 64 units per direction, whose final hidden states are concatenated into a 128-dimensional vector. The **convolutional branch** has three residual blocks with 64, 128 and 256 filters (kernel size 3). Each block contains two convolutions with batch normalization and GELU, plus a Squeeze-and-Excitation module [95] that re-weights the channels through channel-wise attention:

$$s = \sigma(W_2 \text{ReLU}(W_1 \text{GAP}(\mathbf{U}))), \quad \tilde{\mathbf{U}} = s \odot \mathbf{U}. \quad (7.6)$$

The output of the last block is summarized by concatenating global average and global max pooling (512 values). Finally, the two branches are concatenated (640 values) and passed to an MLP (640–320–2) with LayerNorm, GELU and dropout.

Linear layers are initialized with Xavier and convolutions with Kaiming initialization. We train with AdamW [96] (learning rate 10^{-4} , weight decay 10^{-2}) and clip gradients to norm 1. The model has about 0.99 million parameters.

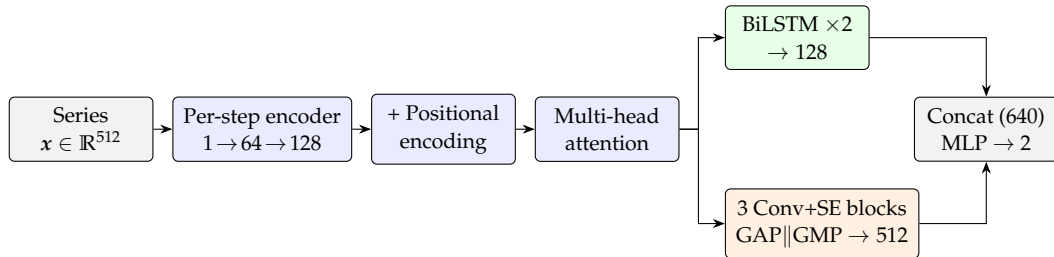


Figure 7.1: The LSTM-FCN variant used in this work.

7.4 MOMENT

7.4.1 Architecture and pre-training

MOMENT [24] is a family of open-source foundation models for time series, pre-trained on a large and varied collection of public series called the Time Series Pile, which spans healthcare, finance, weather, industrial systems and more. The idea is to have one model that, with minimal adaptation, performs well on forecasting, classification, anomaly detection and imputation alike.

The architecture is derived from the encoder of the T5 Transformer [64]. Each time series first goes through **reversible instance normalization** (RevIN), which removes its mean and variance so that the model is insensitive to scale. The series, of fixed length $L = 512$, is then cut into $N = 64$ non-overlapping **patches** of length $P = 8$, following PatchTST [97]. Each patch is linearly projected into a D -dimensional space and treated as a token. The token sequence passes through a stack of Transformer blocks (24 layers with $D = 1024$ in the MOMENT-1-large model used), and finally a task-specific head produces the output. Pre-training uses *masked reconstruction*: some patches are hidden and the model learns to rebuild them from the context. Since this is a self-supervised task that needs no labels, the model is pushed to learn general representations of time series structure.

7.4.2 Option 1: full fine-tuning

The first way to use MOMENT is in its classification configuration, training the whole model (encoder and classification head) on our task and updating every parameter. In a federated setting, however, this means sending hundreds of millions of parameters per client at every round, with a very high communication and local compute cost.

7.4.3 Option 2: frozen embeddings and a classification head

The second option, and one of the main contributions of this thesis, is to use MOMENT frozen, purely as a feature extractor (Figure 7.2). Each client computes, once and locally, the embedding of each of its series: the model runs in embedding mode and the representations of the 64 patches are averaged (mean pooling) into a vector $e \in \mathbb{R}^{1024}$. A lightweight **classification head**, a two-layer MLP, is then trained on these embeddings:

$$h = \text{Dropout}_{0.3}(\text{ReLU}(\text{BN}(W_1 e + b_1))), \quad \hat{y} = W_2 h + b_2, \quad (7.7)$$

$$W_1 \in \mathbb{R}^{512 \times 1024}, \quad W_2 \in \mathbb{R}^{2 \times 512}. \quad (7.8)$$

Only this head takes part in federated training.

The head has about 0.53 million parameters, almost three orders of magnitude fewer than the full model, and this has several advantages in FL. The **communication cost** per round drops accordingly. On the client side, the **computation** boils

down to one forward pass of MOMENT per sample plus training a small MLP, which is realistic for the hardware usually found in a clinical lab. The **representations are identical across clients**, because the extractor is frozen and shared, so part of the heterogeneity between clients is absorbed by a common, stable representation space. Finally, the head is a small differentiable model, well suited to efficient explanation with DeepSHAP (Section 5.6).

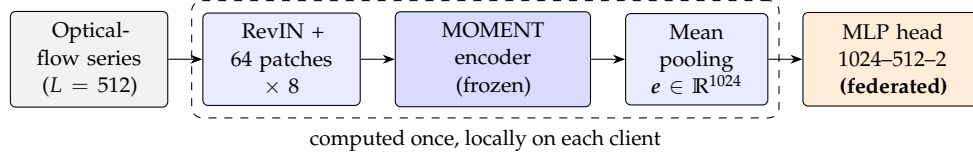


Figure 7.2: The MOMENT-embedding pipeline: the pre-trained encoder is frozen and run locally; only the classification head takes part in federated training.

7.5 Summary

Table 7.1 summarizes the main characteristics of the models and of the parameters exchanged at each communication round.

Table 7.1: Model characteristics and federated parameters.

Model	Approach	Federated parameters	No. of parameters
ROCKET	Random kernels + linear classifier	Logistic regression coefficients	$\approx 20\,000$
ConvTran	Convolution + Transformer	Whole model	$\approx 0.49\text{ M}$
LSTM-FCN	Recurrent + convolutional + attention	Whole model	$\approx 0.99\text{ M}$
MOMENT (fine-tuning)	Foundation model	Whole model	$\approx 3.8 \times 10^8$
MOMENT (embeddings)	Frozen foundation model + MLP	Classification head only	$\approx 0.53\text{ M}$

Chapter 8

Implementation

This chapter explains how the federated systems have been implemented, how the experiments have been run, which metrics and statistical tests have been used to evaluate them, and how SHAP-FL was configured.

8.1 Software

The project is written in Python 3.12. Using PyTorch 2.8 for the neural models, `momentfm` for MOMENT, `sktime` and `scikit-learn` for ROCKET, Flower (≥ 1.28) for federated orchestration, and the `shap` library for explanations. The code is organized in numbered modules that follow the pipeline: data preparation (year-based partitioning, bootstrap analysis, quality checks); one Flower application per model, made of `server_app.py`, `client_app.py` and `task.py`, the last of which holds the model definition and the data loading, training and evaluation functions; the model and experiment configuration files; the scripts for centralized training and for testing on the global set; and finally the results and statistical analyses.

8.1.1 ServerApp

The `ServerApp` initializes the global model, serializes its parameters into an `ArrayRecord` and starts the FedAvg strategy with `fraction-evaluate = 1`, so that every client takes part in every round. In each round the server sends the current parameters and the learning rate (through a `ConfigRecord`) to the clients, collects the updated parameters and aggregates them as in Equation (4.4), weighting each client by the number of training samples it reports. A server-side evaluation function computes the accuracy and loss of the global model on the global test set after each round.

8.1.2 ClientApp

Each ClientApp instance is tied to one partition (partition-id) and implements two operations. The train operation loads the local data, initializes the model with the parameters received from the server, runs E local epochs and sends back the updated parameters together with the training loss and the number of samples.

Local dataset of each client is divided in a train subset (70% of embryos drawn at random) and a validation subset (the remaining 30% of the data).

The evaluate operation evaluates the global model received on the local validation subset and returns loss and accuracy. In the MOMENT-embedding pipeline the ClientApp works directly on the embeddings precomputed and stored locally for each partition (Section 7.4.3), so MOMENT itself never runs during the federated rounds.

8.2 Experimental protocol

The main goal is to measure how federated and centralized training compare as the number of participating centers changes on the first N partitions for every architecture. For each number of partitions $N \in \{2, \dots, 7\}$ the corresponding dataset is taken and the model is trained in two ways. In the **federated** scenario, N clients train the model with FedAvg. In the **centralized** scenario, the same N partitions are merged into a single dataset and the model is trained in the usual way. In both cases the final model is evaluated on the global test set (741 embryos: 407 blastocysts and 334 non-blastocysts). Figure 8.1 illustrates the setup.

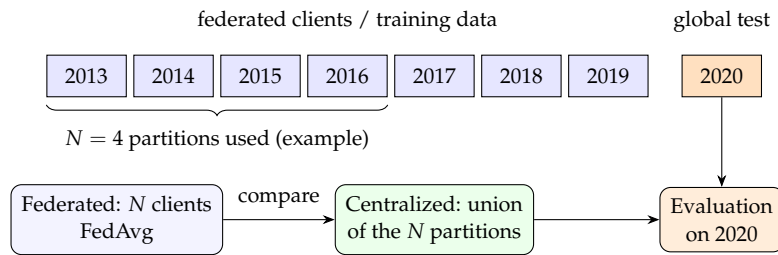


Figure 8.1: Experimental protocol: for each $N \in \{2, \dots, 7\}$ the corresponding partitions are used to train the model in federated and centralized mode; both models are evaluated on the global test set.

This design has two useful properties. As N grows, both the number of clients and the total amount of data increase, so we can see whether the federated model benefits from extra data as much as the centralized one. And since the test set never changes, results from different configurations can be compared directly.

8.3 Evaluation metrics

Let TP , TN , FP and FN be the numbers of true positives, true negatives, false positives and false negatives, with “blastocyst” as the positive class. We use the following metrics:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (8.1)$$

$$\text{Sensitivity (recall)} = \frac{TP}{TP + FN}, \quad \text{Specificity} = \frac{TN}{TN + FP} \quad (8.2)$$

$$\text{Precision} = \frac{TP}{TP + FP}, \quad F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (8.3)$$

$$\text{Balanced accuracy} = \frac{1}{2}(\text{Sensitivity} + \text{Specificity}), \quad (8.4)$$

$$\text{MCC} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}. \quad (8.5)$$

The F1-score is computed separately for each class, taking class 0 or class 1 as positive in turn, and the macro-F1 is their mean. The Matthews correlation coefficient (MCC) ranges from -1 to 1 , with 0 corresponding to random guessing. It is particularly informative when a classifier predicts mostly one class, a case in which accuracy can be misleading [98].

8.4 Statistical analysis

To check whether the training mode (federated or centralized) and the architecture significantly affect accuracy, a **two-way repeated-measures ANOVA** with *mode* (2 levels) and *model* (5 levels) as within-subject factors has been used. The number of partitions N (6 levels) plays the role of the “subject”, i.e. the experimental unit on which all factor combinations are observed. This allows us to look at the main effect of the training mode, the main effect of the architecture, and the mode $\times$ model interaction, which tells us whether the effect of federation depends on the architecture.

In addition, for each model the Wilcoxon signed-rank test has been applied to the paired differences between federated and centralized accuracy. The Friedman test to compare the five architectures within each mode has been applied. The significance level is $\alpha = 0.05$.

8.5 SHAP-FL configuration

SHAP-FL (Section 5.6) has been applied to the global LSTM-FCN model trained with federated learning on $K = 7$ clients, the configuration with the most training data. Each client uses the same model as in the federated rounds, with the same 512-dimensional input: patient identifier, maternal age and the 510 samples of the optical-flow series from 0 h to 127.25 h. After splitting its partition as in the ClientApp (Section 8.1.2), each client draws a background $\mathcal{B}_k$ of 50 samples from its local training set and a set $\mathcal{X}_k$ of 200 samples to explain from its local validation set, both at random (seed 42).

The output explained is the logit $z_1(\mathbf{x})$ of the positive class (blastocyst formation), so the attributions are measured in logit units. SHAP values are computed with `shap.GradientExplainer` with its default of 200 samples per instance; each client then computes its local importance vector $\mathbf{I}^{(k)} \in \mathbb{R}^{512}$, which is the only thing sent to the server.

On server side, the weights m_k in Equation (5.7) are set to the number of local training samples, as in FedAvg. Agreement between clients has been analyzed with the measures described in Section 5.6.2: Spearman correlation between the importance vectors of the time points, Jaccard index on the 20 most important time points and on the 10 most important two-hour windows, and the concentration curve.

Chapter 9

Results

This chapter presents the experimental results. It first compares federated and centralized training for the five models as the number of partitions grows; then it looks in detail at the seven-client configuration and at the statistical tests; finally, it discusses what SHAP-FL tells. All metrics are computed on the global test set (741 embryos, 54.9% blastocysts), which was never used during training. As a point of reference, a trivial classifier that always predicts the majority class would get an accuracy of 0.549 and an MCC of zero on this set.

9.1 Federated vs centralized training

Table 9.1 gives the accuracy of each model for every number of partitions N in both modes, and Figure 9.1 plots the same numbers.

Table 9.1: Accuracy on the global test set as the number of partitions N varies. F = federated (FedAvg), C = centralized, FT = Fine Tuning, emb = Embeddings. The best value in each row and mode is in bold.

N	MOMENT (FT)		LSTM-FCN		ConvTran		ROCKET		MOMENT (emb.)	
	F	C	F	C	F	C	F	C	F	C
2	0.56	0.54	0.84	0.84	0.57	0.56	0.58	0.60	0.61	0.76
3	0.61	0.64	0.86	0.84	0.56	0.56	0.58	0.59	0.70	0.57
4	0.68	0.71	0.86	0.84	0.56	0.78	0.59	0.59	0.77	0.74
5	0.69	0.70	0.86	0.84	0.56	0.74	0.59	0.61	0.70	0.80
6	0.71	0.72	0.87	0.84	0.61	0.74	0.60	0.62	0.81	0.70
7	0.73	0.74	0.85	0.83	0.68	0.82	0.60	0.65	0.81	0.82
Mean	0.663	0.675	0.857	0.838	0.590	0.700	0.590	0.610	0.733	0.732
Std. dev.	0.065	0.074	0.010	0.004	0.048	0.112	0.009	0.023	0.078	0.090

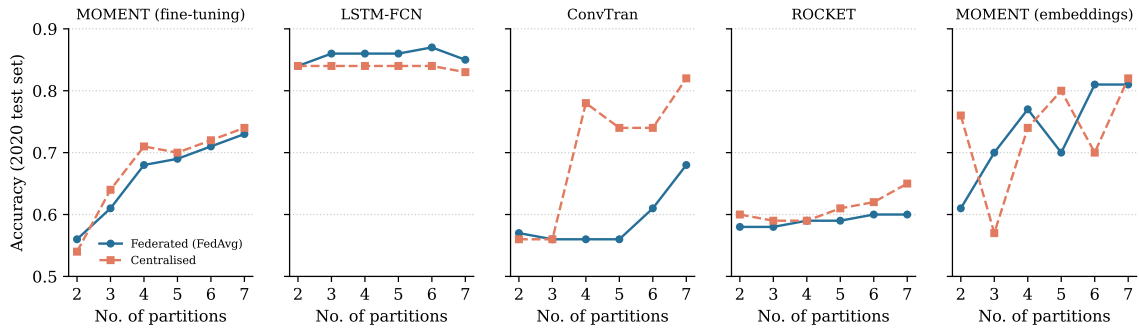


Figure 9.1: Accuracy on the global test set as a function of the number of partitions, for each model, in federated (solid line) and centralized (dashed line) training.

Four distinct patterns emerge.

LSTM-FCN. LSTM-FCN is clearly the best model in both modes, with an accuracy between 0.84 and 0.87 for every N and very little variation (standard deviation 0.010 in federated mode). More interestingly, the federated version is as good as or better than the centralized one in all six configurations. Not being allowed to share the data cost us nothing here. A plausible explanation is that FedAvg acts as a regularizer: by averaging models trained on slightly different distributions, it keeps the global model from fitting the quirks of any single year too closely. Centralized training, run for 50 epochs on the pooled data, reveal the robustness of the model that keeps the same performances on different quantity of data.

MOMENT embeddings. The model built on MOMENT embeddings reaches 0.81 accuracy with 6 and 7 clients, and its best centralized result, 0.82, comes with 7 partitions. Accuracy generally improves with N , but not monotonically, and the swings are large: 0.77 with $N = 4$ but 0.70 with $N = 5$ in federated mode, and a dip to 0.57 with $N = 3$ in centralized mode. These swings suggest that the classification head is somewhat unstable, with predictions that tilt towards one class, as the confusion matrices below confirm. On average the two modes are equivalent (0.733 vs 0.732). This makes it the second-best model overall, and it gets there while federating only about half a million parameters.

Fine-tuning MOMENT. Full fine-tuning of MOMENT improves steadily with the number of partitions (from 0.56 to 0.73 federated, from 0.54 to 0.74 centralized), with little difference between the two modes. Still, it stays well below the same model used as a frozen extractor. Fine-tuning a model with hundreds of millions of parameters on a few thousand examples is very likely sub-optimal: it can damage the pre-trained representations (catastrophic forgetting) without providing enough data to replace them. The frozen option is also far cheaper in a federated setting.

ConvTran and ROCKET. In federated mode ConvTran stays close to the majority-class baseline for $N \leq 5$ (0.56–0.57) and only improves with 6 and 7 clients (0.68). Trained centrally, it goes above 0.74 from $N = 4$ onwards and reaches 0.82 with $N = 7$, which makes it the model with the largest gap between the two modes (11 percentage points on average) probably due to the fact that in federated mode each client has a small dataset, not big enough to let ConvTran learn well. ROCKET gives modest but very stable results in both modes (0.58–0.65), with a small edge for centralized training; it can be considered as the baseline.

Figure 9.2 summarizes the differences between the two modes. The blue cells, where federated training wins, are concentrated on LSTM-FCN, while the deepest red ones belong to ConvTran.

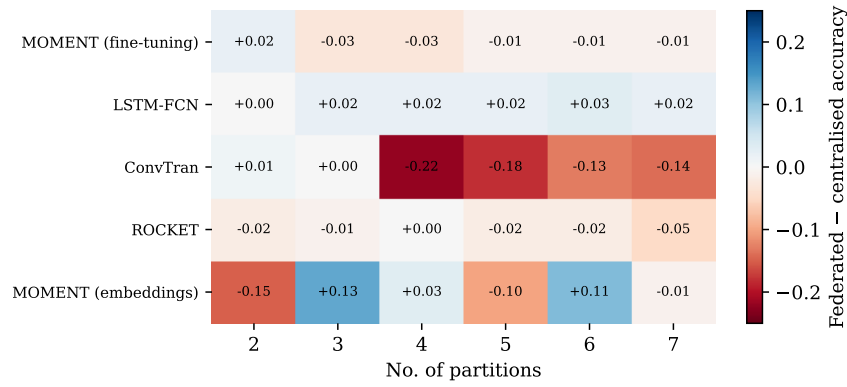


Figure 9.2: Accuracy difference between federated and centralized training for each model and number of partitions. Positive values (blue) indicate higher accuracy for federated training.

9.2 Metrics

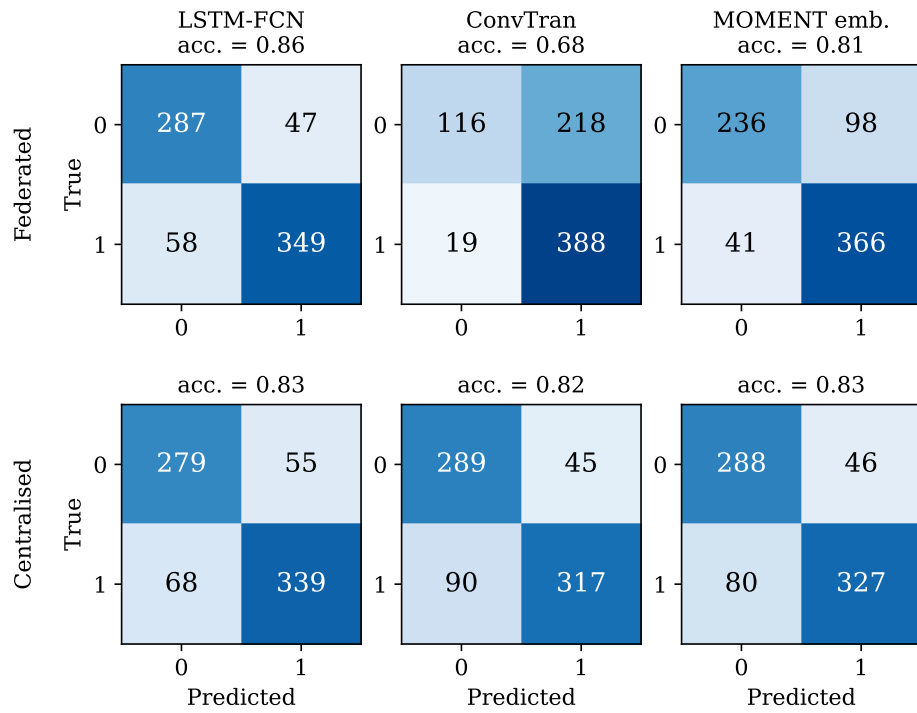


Figure 9.3: Confusion matrices on the global test set ($N = 7$) for LSTM-FCN and MOMENT-embedding in both modes (0 = no blastocyst, 1 = blastocyst).

Federated **LSTM-FCN** is remarkably *balanced*: its sensitivity (0.857) and specificity (0.859) are almost identical, and its MCC, reaching 0.75, is the highest of all configurations.

Federated **MOMENT-embedding** prefers sensitivity (0.899) over specificity (0.707), whereas the centralized version is more balanced. This tilt fits the prior shift described in Chapter 6: FedAvg weights clients by size, and the more recent, larger partitions also have a higher share of blastocysts. The decision threshold could be adjusted afterwards anyway, depending on whether a clinic cares more about not discarding viable embryos (sensitivity) or about not transferring embryos that will not develop (specificity).

Federated **ConvTran** labels almost every embryo as a blastocyst (sensitivity 0.953, specificity 0.347). Its 0.68 accuracy is therefore misleading; the MCC of 0.388 gives a much more honest picture, and this is a good example of why metrics that are sensitive to class prevalence matter.

Figure 9.4 shows the MCC as a function of the number of partitions for the three models with full confusion matrices. Federated LSTM-FCN rises from 0.69 to 0.75 up to $N = 6$ and always stays above its centralized counterpart, while the curves for MOMENT-embedding and ConvTran show the same instability already seen in the accuracy.

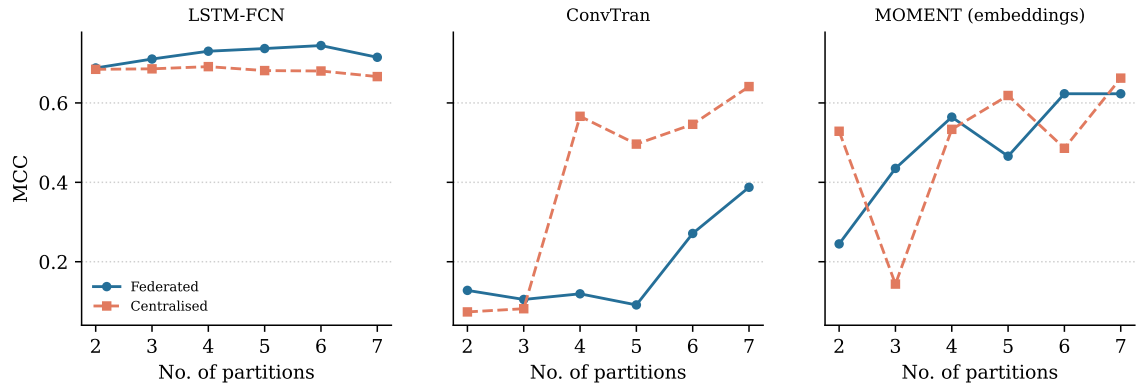


Figure 9.4: Matthews correlation coefficient on the 2020 test set as a function of the number of partitions.

9.3 Statistical analysis

9.3.1 Repeated-measures ANOVA

Table 9.2 reports the repeated-measures ANOVA on accuracy, with *mode* and *model* as within-subject factors and the number of partitions as the subject.

The **model** effect is highly significant ($F = 40.55$, $p < 0.0001$): the choice of architecture is the most important factor. The main effect of the **mode** is not significant ($F = 4.29$, $p = 0.093$); across all models there is no evidence that

Table 9.2: Repeated-measures ANOVA on accuracy (subject: number of partitions, $n = 6$).

Effect	F	Num. df	Den. df	p-value
Mode	4.2914	1	5	0.0931
Model	40.5503	4	20	< 0.0001
Mode $\times$ model	3.3325	4	20	0.0303

federated training systematically loses accuracy compared with centralized training. However, the **interaction**, is significant ($F = 3.33$, $p = 0.030$): the effect of federation depends on the architecture.

9.3.2 Per-model comparisons

Table 9.3 gives, for each model, the mean difference between federated and centralized accuracy and the p-value of the Wilcoxon test on the six pairs. None of the differences is significant at $\alpha = 0.05$.

Table 9.3: Mean accuracy difference (federated – centralized) and Wilcoxon test for each model.

Model	Mean Δ	p-value (Wilcoxon)
MOMENT (fine-tuning)	−0.012	0.250
LSTM-FCN	+0.018	0.063
ConvTran	−0.110	0.125
ROCKET	−0.020	0.063
MOMENT (embeddings)	+0.002	1.000

The Friedman test confirms significant differences among all the architectures and configurations in both federated ($\chi^2 = 20.93$, $p < 0.001$) and centralized mode ($\chi^2 = 15.09$, $p = 0.005$).

9.4 Discussion of performance

9.4.1 LSTM-FCN works best

LSTM-FCN does well because it combines several mechanisms that suit this kind of signal. The convolutional branch with Squeeze-and-Excitation picks up local patterns such as the peaks associated with cell divisions (Figure 6.1); the bidirectional LSTM and the self-attention capture the overall shape of the series and the relations between distant phases of development. With about a million parameters, the model is expressive enough to learn these structures but small enough to train stably on a few hundred examples per client..

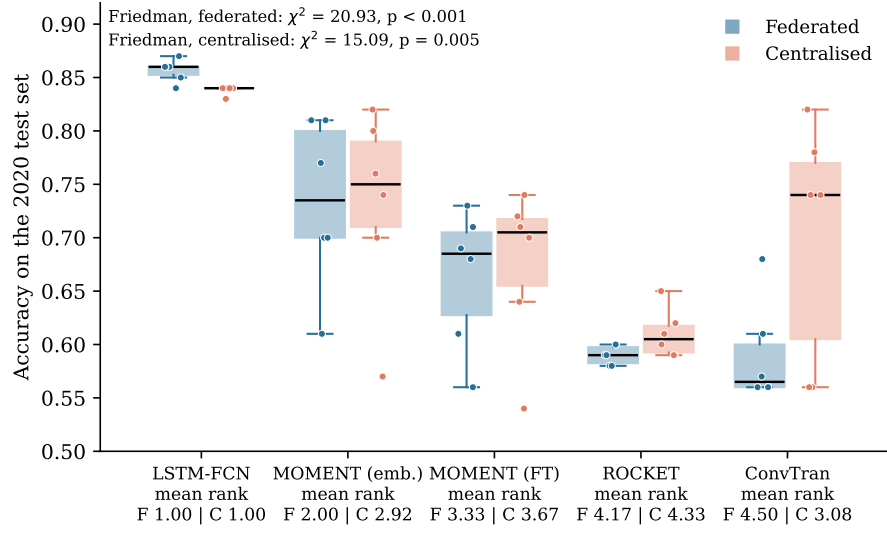


Figure 9.5: Accuracy on the global test set of each architecture in federated (blue) and centralized (orange) mode. Dots are the individual runs. Architectures are ordered by their Friedman mean rank in federated mode (1 = best in every configuration); the mean ranks of both modes are given below each name.

9.4.2 Model size and stability

The two models that suffer most from federation are ConvTran and ROCKET. In federated mode, ConvTran seems stuck, for $N \leq 5$, in a degenerate solution that almost always predicts the positive class.

ROCKET federates the coefficients of a logistic regression on 20 000 features, each trained locally to convergence. Averaging the coefficients of separately trained linear models is not the same as training one linear model on all the data, especially when, as here, the number of features is far larger than the number of samples per client (roughly 300–700 embryos per year).

9.4.3 Embedding approach

From a practical point of view, comparing the two ways of using MOMENT is one of the most useful outcomes of this work. Using the foundation model frozen gives a higher mean accuracy than full fine-tuning (0.733 vs 0.663 in federated mode), reduces the parameters exchanged per round by almost three orders of magnitude, moves the heavy computation (extracting the embeddings) to a single local pass per sample, and yields a model, the MLP head, that DeepSHAP can explain easily. All of this makes the approach well suited to a real network of fertility clinics, where bandwidth, hardware and IT expertise may be limited. The gap with LSTM-FCN (about 4–5 accuracy points at $N = 7$), however, shows that MOMENT’s generic representations, learned from series in very different domains, do not yet fully capture what is specific about the embryo signal. Parameter-efficient adaptation,

for example unfreezing only the last encoder layers, could close part of this gap while keeping communication costs low.

9.5 Federated explainability results

9.5.1 Global importance

Figure 9.6(a) shows the global importance $\bar{I}_j$ of every time point. At the beginning the curve is almost flat; it starts to rise after about 100 h, shows a first peak at around 111 h, drops slightly at about 118 h, and then increases steeply until the end of the series at 127.25 h. Aggregated by day, Figure 9.6 (b) shows that day 5 accounts for 82% of the global importance, meanwhile, days 1 to 4 contribute 3.0%, 3.3%, 5.6% and 6.0%, respectively.

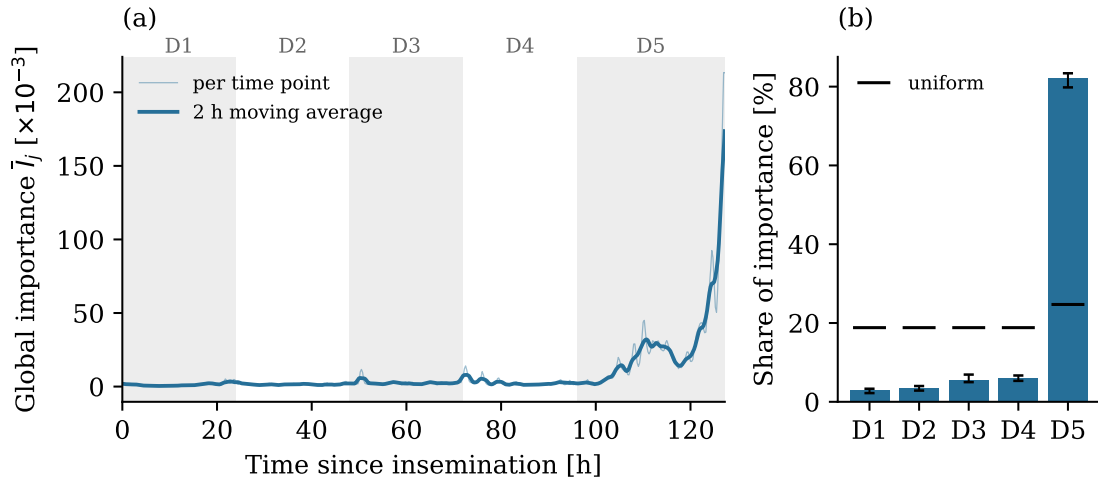


Figure 9.6: Federated global importance of the input time points for the LSTM-FCN model. (a) Importance $\bar{I}_j$ of each time point (thin line) and its two-hour moving average (thick line); the shaded bands mark the days of culture. (b) Share of the total importance falling in each day of culture; the error bars give the range across the seven clients, and the black marks the share each day would have if the importance were uniform over time.

This pattern matches the biology of blastocyst formation. Time of blastocyst formation is about 116 h after insemination [99], and the window in which the importance grows, from about 100 to 127 h, covers cavitation, the appearance of the blastocoel and the expansion of the blastocyst.

Figure 9.7 shows the directions of the attributions. For embryos that reach the blastocyst stage, late time points push the prediction towards the positive class. For the others late time points push it towards the negative class, and in both cases the effect is largest in the last hours. It can be seen that a higher motion during the last

period is associated with a more positive contribution with a correlation between the mean motion and the summed attributions on the last time points of $r = 0.66$.

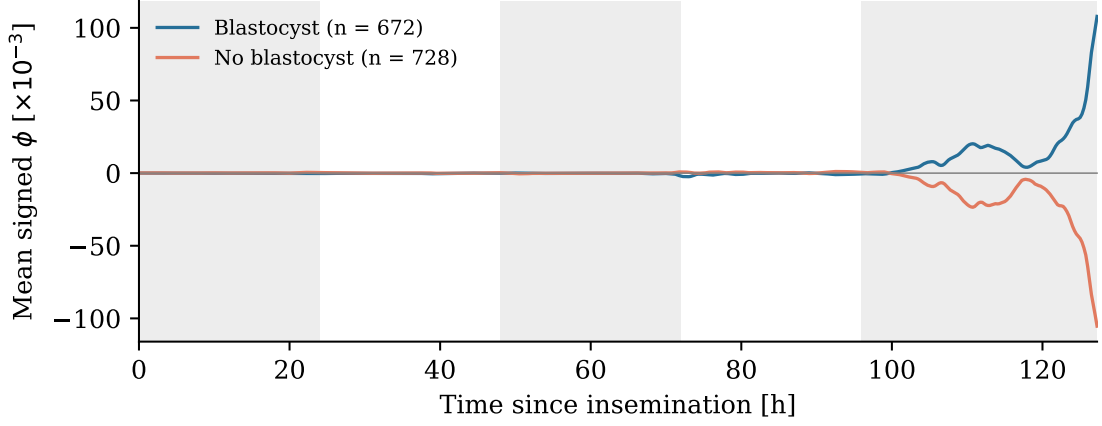


Figure 9.7: Mean signed attribution of each time point (two-hour moving average) for the explained embryos that do (blue) and do not (orange) reach the blastocyst stage over the seven clients. Positive values push prediction towards blastocyst formation.

9.5.2 Agreement between clients

fig. 9.8a compares the local explanation of the clients. The Spearman correlation between the importance vectors of pairs of clients has a mean of 0.95 ranging from 0.91 to 0.98. Each local vector correlates with the global one with ρ between 0.95 and 0.99. In addition, the top features are shared as well: the mean Jaccard index on the 20 most important time points is 0.84 (ranging from 0.74 to 1.00). However,

Furthermore, a weak temporal trend is visible. The mean correlation is 0.96 between clients one or two years apart and decreases to 0.95, 0.94 and 0.91 for gaps of four, five and six years; 2013, the oldest and smallest partition, is the client whose explanations differ most from the others (ρ between 0.91 and 0.94). However, the differences concern the fine ranking of the less important time points rather than the overall pattern: the global model bases its decision on the same phase of development in all the years of the federation.

In conclusion, the global explanation tells us *when*, in development, the model looks. The comparison of the local explanations shows that this behavior is consistent across centers. In a federation of different clinics, a center whose explanations correlated poorly with the others, or whose importance were concentrated in a different phase, would point to differences in acquisition, protocols or population that deserve a closer look.

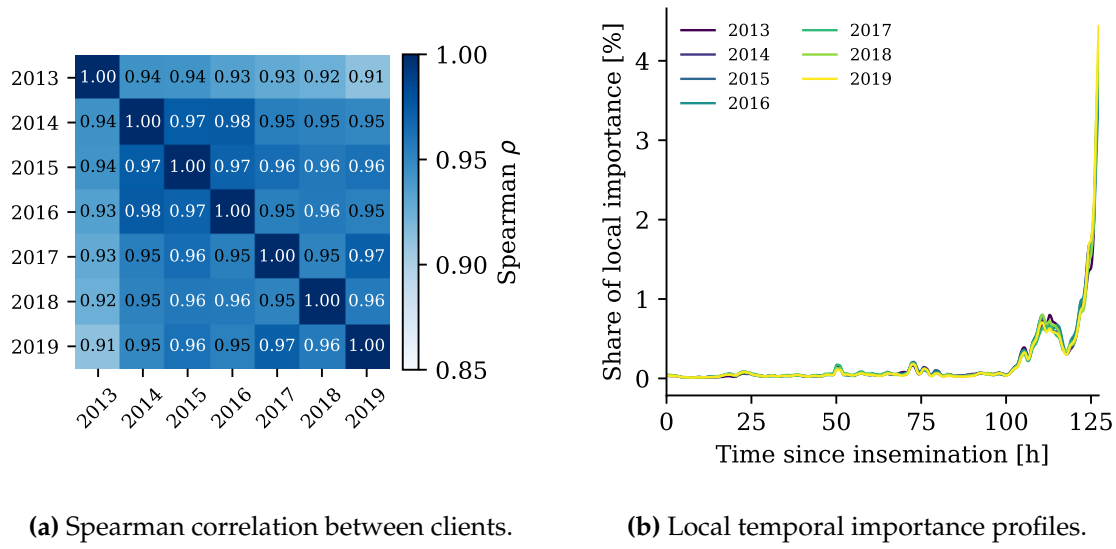


Figure 9.8: Agreement between the local explanations produced by SHAP-FL on the seven clients. (a) Spearman correlation between the local importance vectors of the time points. (b) Local importance of each time point (two-hour moving average), normalized on the total of each client.

9.5.3 Limits

These results should be interpreted with two things in mind.

Firstly, they show that the model relies on the last hours of the series, when cavitation and expansion are already under way. In other words, the model largely recognizes blastocyst formation while it is happening rather than predicting it from early development, which is consistent with the limitation discussed in Section 10.3; training on shorter windows would force the model to use earlier phases, and SHAP-FL would then show whether, and when, early signals are informative. Secondly, the importance of adjacent time points is strongly correlated; the series is smooth and the network processes it sequentially, so individual time points should not be over-interpreted: the reliable information is the time window, not the exact sample.

Chapter 10

Conclusions and future work

10.1 Summary

This thesis explored how Federated Learning and Explainable AI can be applied to predicting blastocyst formation from optical-flow time series extracted from time-lapse videos of human embryos. The starting point was two needs of clinical practice in assisted reproduction: training models on data held by several clinics without pooling them, and making the predictions of those models understandable.

The work makes four main contributions. Firstly, a **realistic federated simulation**, based on splitting the dataset by acquisition year. This introduces natural heterogeneity between clients (label skew and quantity skew) and keeps a whole year, as a global test set that is independent by both year and patient; the maximum number of centers was justified with a bootstrap analysis. Secondly, a **systematic comparison** of five model configurations for time series classification (ROCKET, ConvTran, LSTM-FCN, fine-tuned MOMENT and MOMENT as an embedding extractor) in federated and centralized training, for 2 to 7 clients, implemented with Flower. Thirdly, an **efficient federated pipeline built on a frozen foundation model**, in which only a classification head of about half a million parameters is federated. Fourthly, the **SHAP-FL procedure**, which produces local explanations for each center and an aggregated global explanation without sharing data or per-instance attributions, together with a set of measures to assess how well explanations agree across centers.

10.2 Answers to the research questions

Does federated training cost performance? Not systematically. The repeated-measures ANOVA finds no significant main effect of the training mode ($p = 0.093$), but a significant interaction with the architecture ($p = 0.030$), meaning that the impact of federation depends on the model. For the best model, LSTM-FCN, the federated version matched or beat the centralized one in every configuration, with an accuracy up to 0.87 and an MCC up to 0.75 on the global test set.

Which architecture fits best? LSTM-FCN turned out to be the most accurate and the most stable. The MOMENT-embedding model came second (0.81 accuracy with 7 clients) and offers the best trade-off between accuracy, communication cost and explainability. Fully fine-tuned MOMENT, ConvTran and ROCKET proved less suited to this setting and to the amount of data available.

Can we explain predictions while respecting the federated constraints? Yes. SHAP-FL needs just one extra communication round and sends only aggregated vectors to the server. Applied to LSTM-FCN, it showed that the model bases its decision on the last period of culture, which carries 82% of the total importance and coincides with cavitation and blastocyst expansion. The explanations among the centers agree closely (mean Spearman $\rho = 0.95$), with only a weak decrease as the gap between years grows.

10.3 Limitations

Several limitations should be kept in mind when reading these results; they also point to where the work can be strengthened.

Input time window. The models see the optical-flow series up to about 127 hours after insemination, i.e. into day 5. This window includes compaction and the start of cavitation, when the two classes visibly diverge (Figure 6.1). The task we solved is therefore partly *recognizing* a blastocyst in the making rather than *predicting* it early. For clinical use aimed at early selection, the models should be evaluated on truncated windows (for instance up to day 1, 2 and 3) to see how performance degrades as less of the development is observed.

Tensor layout in ConvTran. In ConvTran implementation, the series reaches the first convolutional layer with the time dimension in the channel position. The model therefore sees the series as a single token with 512 channels, and neither the temporal convolution nor the self-attention works as intended. ConvTran's results should be read with this in mind; it probably hurts its performance, especially in federated mode.

Simulation and single source. The federation is simulated on one machine, and the partitions come from the same laboratory in different years. Heterogeneity between genuinely different clinics, with different equipment, protocols and populations, may be larger than the temporal heterogeneity we studied.

What the explanation reveal. SHAP-FL shows that LSTM-FCN relies almost entirely on the last hours of the series, when the blastocyst is already forming (??).

The explanations are clinically readable, but they also confirm that, with the current input window, the model recognizes blastocyst formation rather than predicting it early.

10.4 Future work

Building on these results and limitations, we see several directions for future work.

Firstly, **early prediction**: training and evaluating the models on progressively shorter time windows, to find out how early blastocyst formation can be predicted. This would support earlier decisions and shorter culture.

Secondly, **study early prediction** using SHAP-FL. Explaining models trained on progressively shorter windows would show whether, and when, earlier phases of development, such as the first cleavages or compaction, carry information about blastocyst formation, and the resulting maps could be compared with the embryologists' morphokinetic annotations.

Thirdly, trying **aggregation strategies designed for non-IID data**, such as Fed-Prox [66], FedOpt [69] and personalized FL [68]. These could help mostly the models that are more sensitive to heterogeneity (ConvTran, ROCKET) and correct the imbalance between sensitivity and specificity caused by the prior shift.

Furthermore, between freezing a foundation model completely and fine-tuning it entirely, there are intermediate options for **efficient adaptation**, such as unfreezing the last encoder layers or using low-rank adapters. They might bring MOMENT closer to LSTM-FCN while keeping communication costs low.

Moreover, on the privacy side, **formal guarantees** could be added by integrating secure aggregation [76] and Differential Privacy [75], both for the model updates and for the SHAP-FL importance vectors, and measuring the trade-off between privacy level and accuracy.

In addition, a **real multi-center validation**, involving several IVF clinics in a genuine federated deployment through Flower's distributed mode, would test how well the models generalize across equipment and populations and whether the approach is sustainable in practice.

Finally, **multimodal integration** could combine the optical-flow series with clinical variables (maternal age, semen quality, culture medium) and morphokinetic annotations, and target clinically more relevant endpoints than blastocyst formation, such as euploidy, implantation and live birth.

10.5 Final remarks

The results show that a federated learning system can reach, for the prediction of embryo development, performance comparable to or better than a centralized system, and that explainability can be built into the federated paradigm without giving up data locality. Turning this into a clinical tool will still require consolidating the methodology, extending the explanations to the time domain and validating the approach on several clinics. We hope this work provides a useful foundation, and some evidence, for taking those next steps.

Bibliography

- [1] Lorenzo Corso. Sviluppo di modelli di intelligenza artificiale per supportare la fecondazione in vitro. Master's thesis, Università degli studi di Pavia, Pavia, 2023/2024.
- [2] Shane Windsor. CellPIV.pdf.
- [3] World Health Organization. Infertility – fact sheet. <https://www.who.int/news-room/fact-sheets/detail/infertility>. Accessed on 02/09/2026.
- [4] Purity Njagi et al. Financial costs of assisted reproductive technology for patients in low- and middle-income countries: a systematic review. *Human Reproduction Open*, 2023(2):hoad007, 2023. doi: 10.1093/hropen/hoad007.
- [5] Carol Coughlan and William L. Ledger. In-vitro fertilisation. *Obstetrics, Gynaecology & Reproductive Medicine*, 18(11):300–306, 2008. doi: 10.1016/j.ogrm.2008.08.009.
- [6] Jack Yu Jen Huang and Zev Rosenwaks. In vitro fertilisation treatment and factors affecting success. *Best Practice & Research Clinical Obstetrics & Gynaecology*, 26(6):777–788, 2012. doi: 10.1016/j.bpobgyn.2012.08.017.
- [7] Rocío Nuñez-Calonge et al. Making and selecting the best embryo in in vitro fertilization. *Archives of Medical Research*, 55(8):103068, 2024. doi: 10.1016/j.arcmed.2024.103068.
- [8] Sebastiaan Mastenbroek et al. Embryo selection in IVF. *Human Reproduction*, 26(5):964–966, 2011. doi: 10.1093/humrep/der050.
- [9] Ashleigh Storr et al. Inter-observer and intra-observer agreement between embryologists during selection of a single day 5 embryo for transfer: a multi-center study. *Human Reproduction*, 32(2):307–314, 2017. doi: 10.1093/humrep/dew330.
- [10] Allison E. Baxter Bendus et al. Interobserver and intraobserver variation in day 3 embryo grading. *Fertility and Sterility*, 86(6):1608–1615, 2006. doi: 10.1016/j.fertnstert.2006.05.037.

- [11] Goedeke Paternot et al. Intra- and inter-observer analysis in the morphological assessment of early-stage embryos. *Reproductive Biology and Endocrinology*, 7 (1):105, 2009. doi: 10.1186/1477-7827-7-105.
- [12] María Cruz et al. Timing of cell division in human cleavage-stage embryos is linked with blastocyst formation and quality. *Reproductive BioMedicine Online*, 25(4):371–381, 2012. doi: 10.1016/j.rbmo.2012.06.017.
- [13] Tammy Lee et al. A brief history of artificial intelligence embryo selection: from black-box to glass-box. *Human Reproduction*, 39(2):285–292, 2024. doi: 10.1093/humrep/dead254.
- [14] Errol R. Norwitz, Valentine Edusa, and Joong Shin Park. Maternal physiology and complications of multiple pregnancy. *Seminars in Perinatology*, 29(5):338–348, 2005. doi: 10.1053/j.semperi.2005.08.002.
- [15] Marianna R. Dare et al. Single or multiple embryo transfer following in vitro fertilisation for improved neonatal outcome: A systematic review of the literature. *Australian and New Zealand Journal of Obstetrics and Gynaecology*, 44 (4):283–291, 2004. doi: 10.1111/j.1479-828X.2004.00243.x.
- [16] Mikkel Fly Kragh and Henrik Karstoft. Embryo selection with artificial intelligence: how to evaluate and compare methods? *Journal of Assisted Reproduction and Genetics*, 38(7):1675–1689, 2021. doi: 10.1007/s10815-021-02254-6.
- [17] Peter Kovacs. Embryo selection: the role of time-lapse monitoring. *Reproductive Biology and Endocrinology*, 12(1):124, 2014. doi: 10.1186/1477-7827-12-124.
- [18] M. VerMilyea et al. Development of an artificial intelligence-based assessment model for prediction of embryo viability using static images captured by optical light microscopy during IVF. *Human Reproduction*, 35(4):770–784, 2020. doi: 10.1093/humrep/deaa013.
- [19] M. Salih et al. Embryo selection through artificial intelligence versus embryologists: a systematic review. *Human Reproduction Open*, 2023(3):hoad031, 2023. doi: 10.1093/hropen/hoad031.
- [20] Helen Y. Yang et al. Blastassist: a deep learning pipeline to measure interpretable features of human embryos. *Human Reproduction*, 39(4):698–708, 2024. doi: 10.1093/humrep/deae024.
- [21] S. Canosa et al. A novel machine-learning framework based on early embryo morphokinetics identifies a feature signature associated with blastocyst development. *Journal of Ovarian Research*, 17(1):63, 2024. doi: 10.1186/s13048-024-01376-6.

-
- [22] Lucia Maresca et al. Artificial intelligence in embryo selection: Current approaches and clinical implications. *Bioengineering*, 13(6):688, 2026. doi: 10.3390/bioengineering13060688.
- [23] Rishi Bommasani et al. On the opportunities and risks of foundation models. arXiv:2108.07258, 2021.
- [24] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *PMLR*, 2024.
- [25] European Parliament and Council of the European Union. Regulation (eu) 2016/679 on the protection of natural persons with regard to the processing of personal data (general data protection regulation). Official Journal of the European Union, L 119, 2016.
- [26] Nguyen Truong, Kai Sun, Siyao Wang, Florian Guitton, and YiKe Guo. Privacy preservation in federated learning: An insightful survey from the GDPR perspective. *Computers & Security*, 110:102402, 2021. doi: 10.1016/j.cose.2021.102402.
- [27] Gustavo Hernández-Peñaloza et al. Federated learning and medical device regulation: Bridging gaps in healthcare AI governance. In *2025 IEEE International Conference on E-health Networking, Application & Services (Healthcom)*, pages 25–31, 2025. doi: 10.1109/HealthCom60686.2025.11342827.
- [28] EU Artificial Intelligence Act – up-to-date developments and analyses of the EU AI Act. <https://artificialintelligenceact.eu/>. Accessed on 27/08/2026.
- [29] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *PMLR*, pages 1273–1282, 2017.
- [30] Peter Kairouz, H. Brendan McMahan, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. doi: 10.1561/22000000083.
- [31] Nicola Rieke et al. The future of digital health with federated learning. *npj Digital Medicine*, 3:119, 2020. doi: 10.1038/s41746-020-00323-1.
- [32] Micah J. Sheller et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10: 12598, 2020. doi: 10.1038/s41598-020-69250-1.

- [33] Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (XAI): Toward medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, 32(11):4793–4813, 2021. doi: 10.1109/TNNLS.2020.3027314.
- [34] Alejandro Barredo Arrieta et al. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, 2020. doi: 10.1016/j.inffus.2019.12.012.
- [35] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, pages 4765–4774, 2017.
- [36] Emily S. Jungheim, Melissa F. Meyer, and Darcy E. Broughton. Best practices for controlled ovarian stimulation in in vitro fertilization. *Seminars in Reproductive Medicine*, 33(2):77–82, 2015. doi: 10.1055/s-0035-1546424.
- [37] Arri Coomarasamy et al. Urinary hMG versus recombinant FSH for controlled ovarian hyperstimulation following an agonist long down-regulation protocol in IVF or ICSI treatment: a systematic review and meta-analysis. *Human Reproduction*, 23(2):310–315, 2008. doi: 10.1093/humrep/dem305.
- [38] Mae Wu Healy, Micah J. Hill, and Eric D. Levens. Optimal oocyte retrieval and embryo transfer techniques: Where we are and how we got here. *Seminars in Reproductive Medicine*, 33(2):83–91, 2015. doi: 10.1055/s-0035-1545365.
- [39] Iván Oseguera-López et al. Novel techniques of sperm selection for improving IVF and ICSI outcomes. *Frontiers in Cell and Developmental Biology*, 7, 2019. doi: 10.3389/fcell.2019.00298.
- [40] Denny Sakkas et al. Sperm selection in natural conception: what can we learn from mother nature to improve assisted reproduction outcomes? *Human Reproduction Update*, 21(6):711–726, 2015. doi: 10.1093/humupd/dmv042.
- [41] C. L. O’Neill et al. Development of ICSI. *Reproduction*, 156(1):F51–F58, 2018. doi: 10.1530/REP-18-0011.
- [42] Yoshiteru Kai et al. Assessment of developmental potential of human single pronucleated zygotes derived from conventional in vitro fertilization. *Journal of Assisted Reproduction and Genetics*, 35(8):1377–1384, 2018. doi: 10.1007/s10815-018-1241-2.
- [43] Mutsuko Fujiwara et al. Effect of micro-environment maintenance on embryo culture after in-vitro fertilization: comparison of top-load mini incubator and conventional front-load incubator. *Journal of Assisted Reproduction and Genetics*, 24(1):5–9, 2007. doi: 10.1007/s10815-006-9088-3.
- [44] Fernando J. Prados et al. The cleavage stage embryo. *Human Reproduction*, 27 (suppl_1):i50–i71, 2012. doi: 10.1093/humrep/des224.

- [45] Giovanni Coticchio et al. The enigmatic morula: mechanisms of development, cell fate determination, self-correction and implications for ART. *Human Reproduction Update*, 25(4):422–438, 2019. doi: 10.1093/humupd/dmz008.
- [46] Kevin S. Richter et al. Quantitative grading of a human blastocyst: optimal inner cell mass size and shape. *Fertility and Sterility*, 76(6):1157–1167, 2001. doi: 10.1016/S0015-0282(01)02870-9.
- [47] David K. Gardner and William B. Schoolcraft. In vitro culture of human blastocysts. In R. Jansen and D. Mortimer, editors, *Towards Reproductive Certainty: Fertility and Genetics Beyond 1999*, pages 378–388. Parthenon Publishing, Carnforth, 1999.
- [48] John D. Aplin. The cell biological basis of human implantation. *Best Practice & Research Clinical Obstetrics & Gynaecology*, 14(5):757–764, 2000. doi: 10.1053/beog.2000.0116.
- [49] K. Hardy. Apoptosis in the human embryo. *Reviews of Reproduction*, 4(3): 125–134, 1999. doi: 10.1530/revreprod/4.3.125.
- [50] Zsolt Peter Nagy, Daniel Shapiro, and Ching-Chien Chang. Vitrification of the human embryo: a more efficient and safer in vitro fertilization treatment. *Fertility and Sterility*, 113(2):241–247, 2020. doi: 10.1016/j.fertnstert.2019.12.009.
- [51] iDAScore-intelligent data analysis for embryo evaluation, . URL <https://www.vitrolife.com/our-products/idascore-intelligent-data-analysis-for-embryo-evaluation/>.
- [52] Jørgen Berntsen, Jens Rimestad, Jacob Theilgaard Lassen, Dang Tran, and Mikkel Fly Kragh. Robust and generalizable embryo selection based on artificial intelligence and time-lapse image sequences. 17(2):e0262661. ISSN 1932-6203. doi: 10.1371/journal.pone.0262661. URL <https://doi.org/10.1371/journal.pone.0262661>.
- [53] Jacob Theilgaard Lassen, Mikkel Fly Kragh, Jens Rimestad, Martin Nygård Johansen, and Jørgen Berntsen. Development and validation of deep learning based embryo selection across multiple days of transfer. 13(1):4235. ISSN 2045-2322. doi: 10.1038/s41598-023-31136-3. URL <https://doi.org/10.1038/s41598-023-31136-3>.
- [54] Satoshi Ueno, Jørgen Berntsen, Motoki Ito, Tadashi Okimura, Akiko Yabuuchi, Keiichi Kato, et al. Pregnancy prediction performance of an annotation-free embryo scoring system on the basis of deep learning after single vitrified-warmed blastocyst transfer: a single-center large cohort retrospective study. *Fertility and Sterility*, 116(4):1172–1180, 2021. doi: 10.1016/j.fertnstert.2021.06.001.

- [55] Jiahong Zhu, Lihong Wu, Junxia Liu, others, Xiangwei Hao, Guoning Huang, and Wei Han. External validation of a model for selecting day 3 embryos for transfer based upon deep learning and time-lapse imaging. *Reproductive BioMedicine Online*, 47(3):103242, 2023. doi: 10.1016/j.rbmo.2023.103242.
- [56] Danilo Cimadomo, Viviana Chiappetta, Federica Innocenti, Gaia Saturno, Marilena Taggi, Anabella Marconetto, Valentina Casciani, Laura Albricci, Roberta Maggiulli, Giovanni Coticchio, Aisling Ahlström, Jørgen Berntsen, Mark Larman, Andrea Borini, Alberto Vaiarelli, Filippo Maria Ubaldi, and Laura Rienzi. Towards automation in IVF: pre-clinical validation of a deep learning-based embryo grading system during PGT-A cycles. *Journal of Clinical Medicine*, 12(5):1806, 2023. doi: 10.3390/jcm12051806.
- [57] Wei Han, Bo Huang, Jiahong Zhu, et al. Correlation of fetal heartbeat outcome after Day 3 or Day 5 single embryo transfer of morphologically selected embryos with an annotation-free deep learning scoring system: results from a multi-center study. *Journal of Assisted Reproduction and Genetics*, 42:3397–3407, 2025. doi: 10.1007/s10815-025-03570-x.
- [58] Martin N. Johansen, Erik T. Parner, Mikkel F. Kragh, Keiichi Kato, Satoshi Ueno, Stefan Palm, Manuel Kernbach, Başak Balaban, İpek Keleş, Anette V. Gabrielsen, Lea H. Iversen, and Jørgen Berntsen. Comparing performance between clinics of an embryo evaluation algorithm based on time-lapse images and machine learning. *Journal of Assisted Reproduction and Genetics*, 40(9):2129–2137, 2023. doi: 10.1007/s10815-023-02871-3.
- [59] Peter J. Illingworth, Christos Venetis, David K. Gardner, Scott M. Nelson, Jørgen Berntsen, Mark G. Larman, Franca Agresta, Saran Ahitan, Aisling Ahlström, Fleur Cattrall, Simon Cooke, Kristy Demmers, Anette Gabrielsen, Johnny Hindkjær, Rebecca L. Kelley, Charlotte Knight, Lisa Lee, Robert Lahoud, Manveen Mangat, Hannah Park, Anthony Price, Geoffrey Trew, Bettina Troest, Anna Vincent, Susanne Wennerström, Lyndsey Zujovic, and Thorir Hardarson. Deep learning versus manual morphology-based embryo selection in IVF: a randomized, double-blind noninferiority trial. 30(11). ISSN 1078-8956. doi: 10.1038/s41591-024-03166-5. URL <https://pmc.ncbi.nlm.nih.gov/articles/PMC11564097/>.
- [60] Andrea Park. Fairtility welcomes EU approval for embryo assessment AI. URL <https://www.fiercebiotech.com/medtech/fairtility-welcomes-eu-approval-embryo-assessment-ai>.
- [61] Decoding embryo development: the effect of clinical variables in morphokinetics and artificial intelligence quality scoring - reproductive BioMedicine online, . URL [https://www.rbmojournal.com/article/S1472-6483\(25\)00073-2/fulltext](https://www.rbmojournal.com/article/S1472-6483(25)00073-2/fulltext).

- [62] U.S. Food and Drug Administration. 510(k) premarket notification K243851: CHLOE Blast (Fairtility Ltd.). <https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfPMN/pmn.cfm?ID=K243851>, 2025. Decision date 15 August 2025; accessed October 2026.
- [63] Fairtility Ltd. Pivotal observational CHLOE study. ClinicalTrials.gov identifier NCT06111365, <https://clinicaltrials.gov/study/NCT06111365>, 2023. Accessed October 2026.
- [64] Colin Raffel et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.
- [65] Johan Sveen. An introduction to MatPIV v. 1.6.1.
- [66] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems (MLSys)*, volume 2, pages 429–450, 2020.
- [67] Cina Mathew and P. Asha. Fedprox: Fedsplit algorithm based federated learning for statistical and system heterogeneity in medical data communication. *Journal of Internet Services and Information Security*, 14(3):353–370, 2024. doi: 10.58346/JISIS.2024.I3.021.
- [68] Alysa Ziyang Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12):9587–9603, 2023. doi: 10.1109/TNNLS.2022.3160699.
- [69] Sashank J. Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations (ICLR)*, 2021.
- [70] Syed Thouheed Ahmed et al. Fedopt: federated learning-based heterogeneous resource recommendation and optimization for edge computing. *Soft Computing*, 2024. doi: 10.1007/s00500-023-09542-6.
- [71] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-IID data. arXiv:1806.00582, 2018.
- [72] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-IID data silos: An experimental study. In *IEEE 38th International Conference on Data Engineering (ICDE)*, pages 965–978, 2022.

- [73] Georgios A. Kaissis, Marcus R. Makowski, Daniel Rückert, and Rickmer F. Braren. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 2:305–311, 2020. doi: 10.1038/s42256-020-0186-1.
- [74] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.
- [75] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4):211–407, 2014. doi: 10.1561/04000000042.
- [76] Keith Bonawitz et al. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1175–1191, 2017. doi: 10.1145/3133956.3133982.
- [77] Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchu Qiu, Javier Fernandez-Marques, Yan Gao, Lorenzo Sani, Kwing Hei Li, Titouan Parcollet, Pedro Porto Buarque de Gusmão, and Nicholas D. Lane. Flower: A friendly federated learning research framework. arXiv:2007.14390, 2020.
- [78] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. arXiv:1702.08608, 2017.
- [79] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, 2019. doi: 10.1038/s42256-019-0048-x.
- [80] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “why should I trust you?”: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016. doi: 10.1145/2939672.2939778.
- [81] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of PMLR, pages 3145–3153, 2017.
- [82] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of PMLR, pages 3319–3328, 2017.
- [83] Lloyd S. Shapley. A value for n-person games. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games, Volume II*, number 28 in Annals of Mathematics Studies, pages 307–317. Princeton University Press, 1953.

-
- [84] Scott M. Lundberg et al. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2:56–67, 2020. doi: 10.1038/s42256-019-0138-9.
- [85] Christoph Düsing and Philipp Cimiano. SHAP–FL: Improving explainability for multicentric sepsis onset prediction through background dataset synthesis. In Riccardo Bellazzi, José Manuel Juárez Herrero, Lucia Sacchi, and Blaž Zupan, editors, *Artificial Intelligence in Medicine*, pages 120–129. Springer Nature Switzerland. ISBN 9783031958380. doi: 10.1007/978-3-031-95838-0_12.
- [86] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Image Analysis (SCIA 2003)*, volume 2749 of *Lecture Notes in Computer Science*, pages 363–370. Springer, 2003. doi: 10.1007/3-540-45103-X_50.
- [87] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 674–679, 1981.
- [88] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33:917–963, 2019. doi: 10.1007/s10618-019-00619-1.
- [89] Angus Dempster, François Petitjean, and Geoffrey I. Webb. ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34:1454–1495, 2020. doi: 10.1007/s10618-020-00701-z.
- [90] Navid Mohammadi Foumani, Chang Wei Tan, Geoffrey I. Webb, and Mahsa Salehi. Improving position encoding of transformers for multivariate time series classification. *Data Mining and Knowledge Discovery*, 38:22–48, 2024. doi: 10.1007/s10618-023-00948-2.
- [91] Ashish Vaswani et al. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, 2017.
- [92] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. LSTM fully convolutional networks for time series classification. *IEEE Access*, 6: 1662–1669, 2018. doi: 10.1109/ACCESS.2017.2779939.
- [93] Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1578–1585, 2017. doi: 10.1109/IJCNN.2017.7966039.
- [94] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

- [95] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7132–7141, 2018.
- [96] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *International Conference on Learning Representations (ICLR)*, 2019.
- [97] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations (ICLR)*, 2023.
- [98] Davide Chicco and Giuseppe Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21:6, 2020. doi: 10.1186/s12864-019-6413-7.
- [99] Alpha Scientists in Reproductive Medicine and ESHRE Special Interest Group of Embryology. The Istanbul consensus workshop on embryo assessment: proceedings of an expert meeting. *Human Reproduction*, 26(6):1270–1283, 2011. doi: 10.1093/humrep/der037.