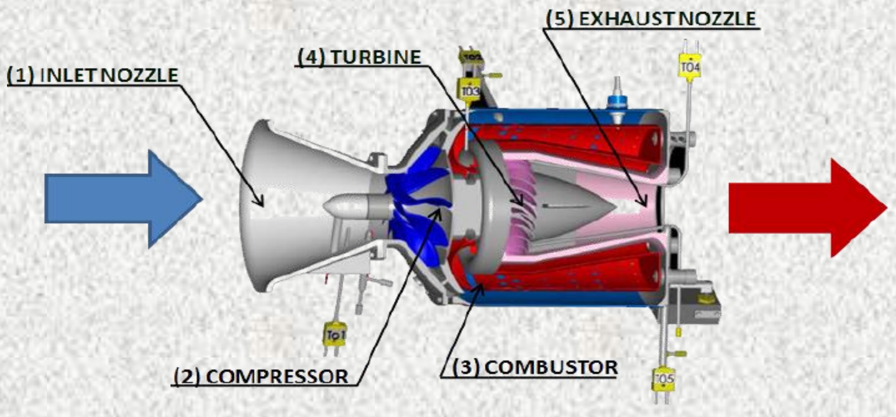
Indian Institute of Technology Bombay

Dynamic Modeling Of Mini SR-30 Gas Turbine Engine



Presented By:

Prof. P. S. V. Nataraj

Outline

- Quick glance at deep learning
- Introduction to gas turbine engine
- First principle based modeling
- Deep learning based modeling
- Results and validation
- Conclusion

Quick Glance at Deep Learning

A brief Introduction:

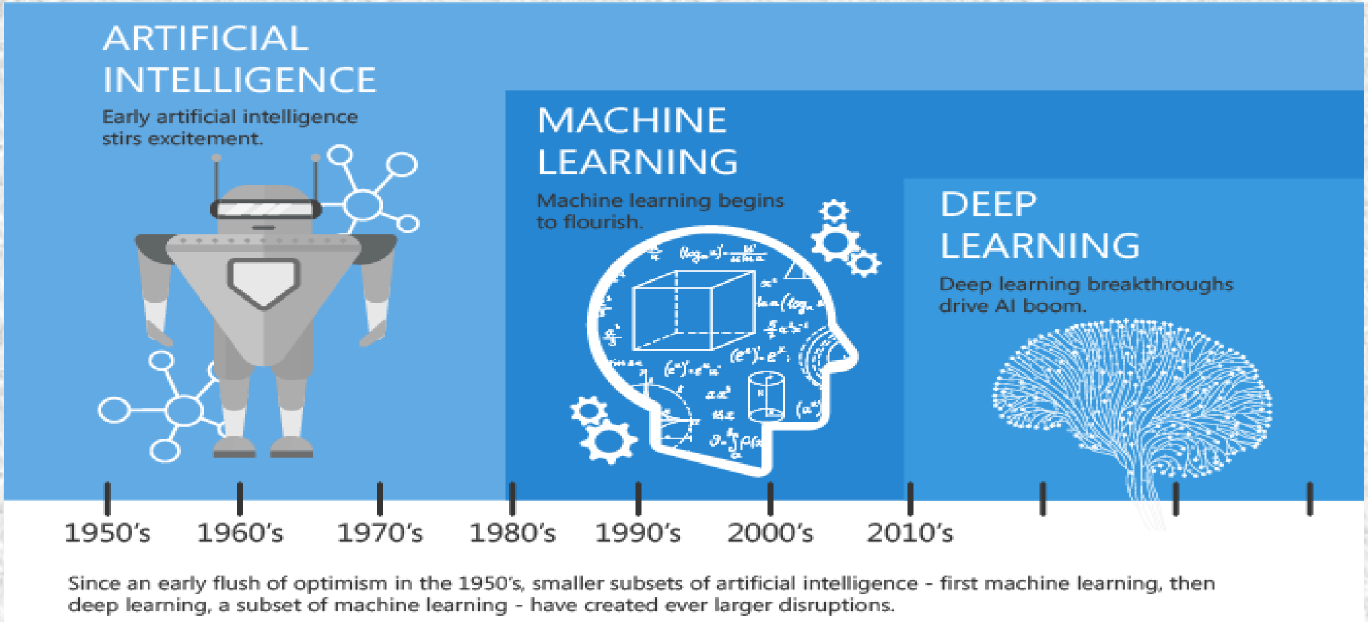
- **1943** - Walter Pitts and Warren McCulloch, gave us that piece of the puzzle when they created the **first mathematical model** of a neural network.
- **1946 –** John Mauchly & J. Presper Eckert develop world's first digital computer **'ENIAC'.**
- **1952** — Arthur Samuel writes the **first computer program** capable of learning.
- **1958** — Frank Rosenblatt designs the **Perceptron**, the first artificial neural network.

What has fueled the development of deep learning?

1. Explosion of data.
2. Cheap computing cost – CPUs and GPUs.
3. Improvement of machine learning models.

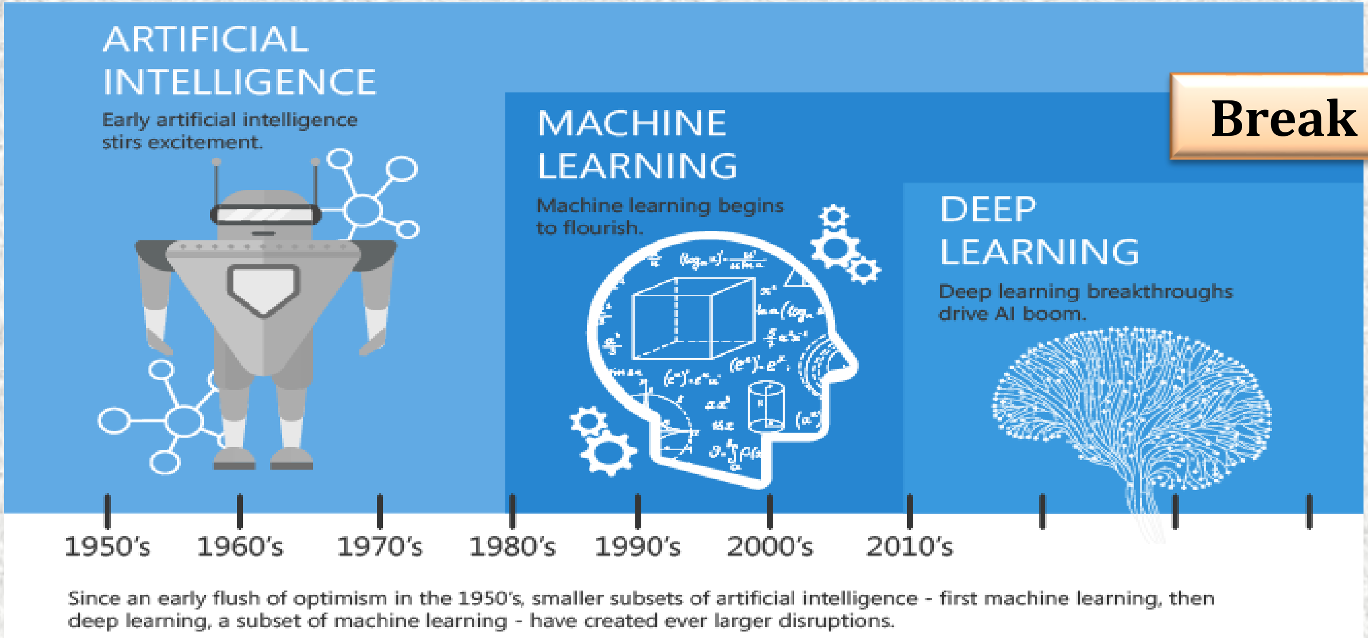
Quick Glance at Deep Learning

Evolution of Deep Learning



Quick Glance at Deep Learning

Evolution of Deep Learning



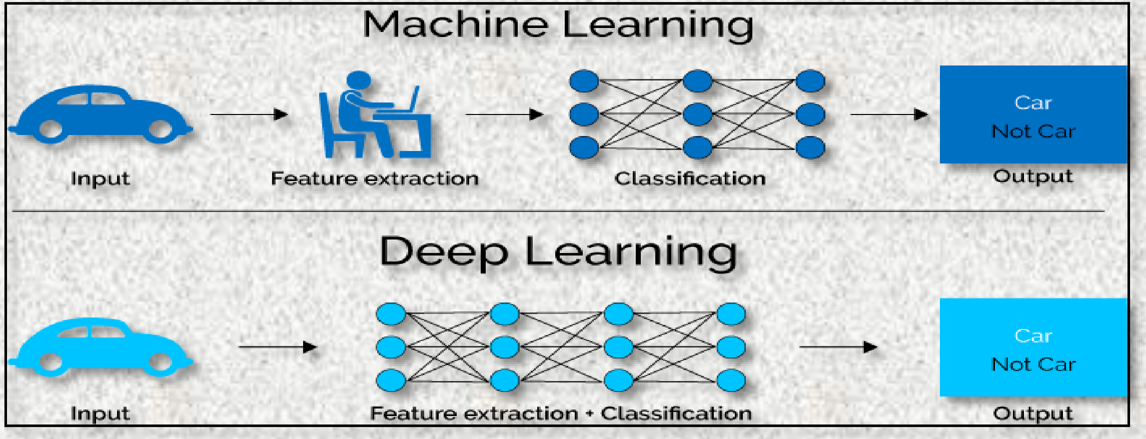
Break through

- **Algorithm**
- **Big data**
- **CPU/GPU**

Quick Glance at Deep Learning

Deep Learning revolutionized Machine learning:

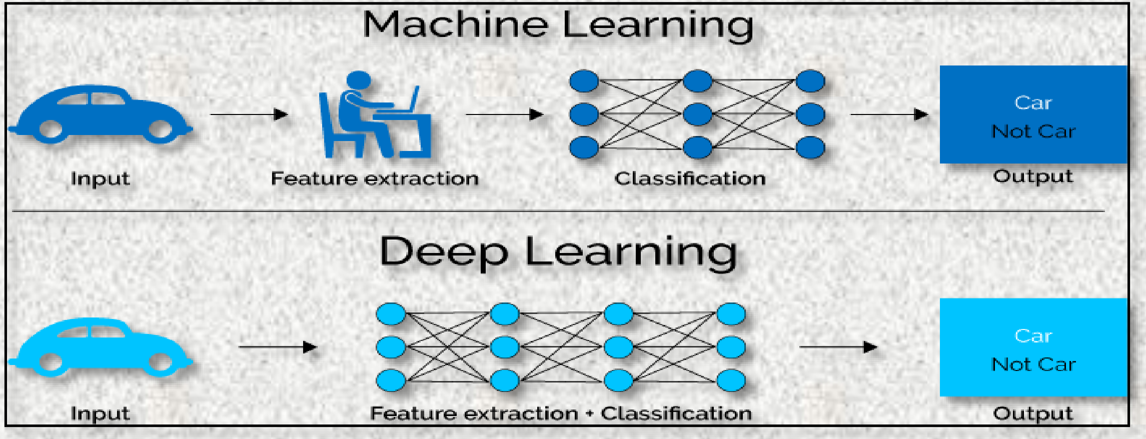
- Deep learning don't need to provide features ahead of time, it learns features at different levels by itself.
- Same deep learning architecture can be trained to accomplish different tasks.

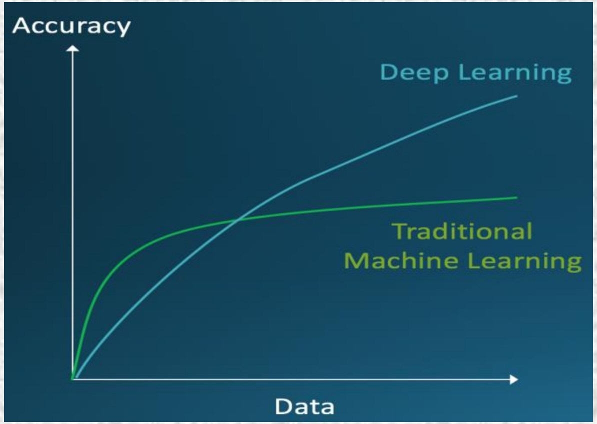


Quick Glance at Deep Learning

Deep Learning revolutionized Machine learning:

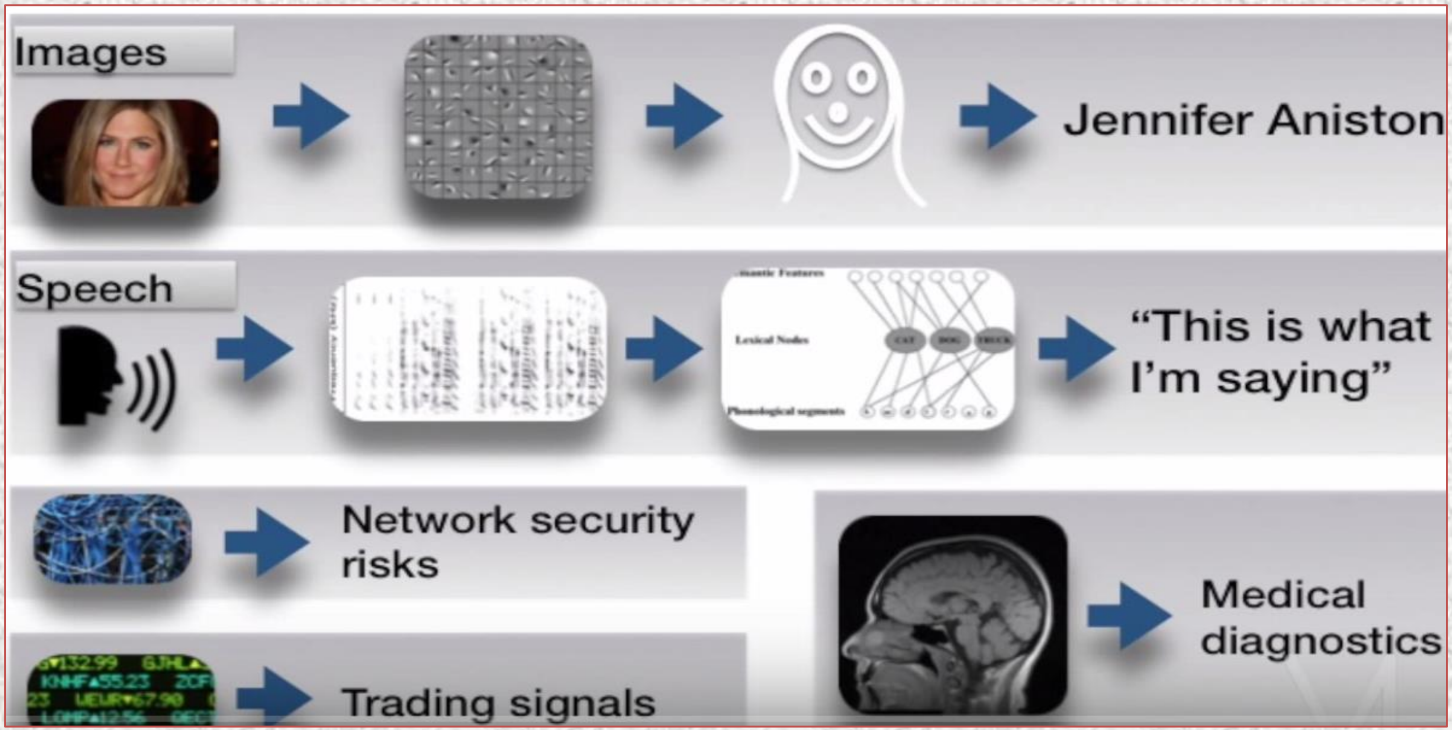
- Deep learning don’t need to provide features ahead of time, it learns features at different levels by itself.
- Same deep learning architecture can be trained to accomplish different tasks.





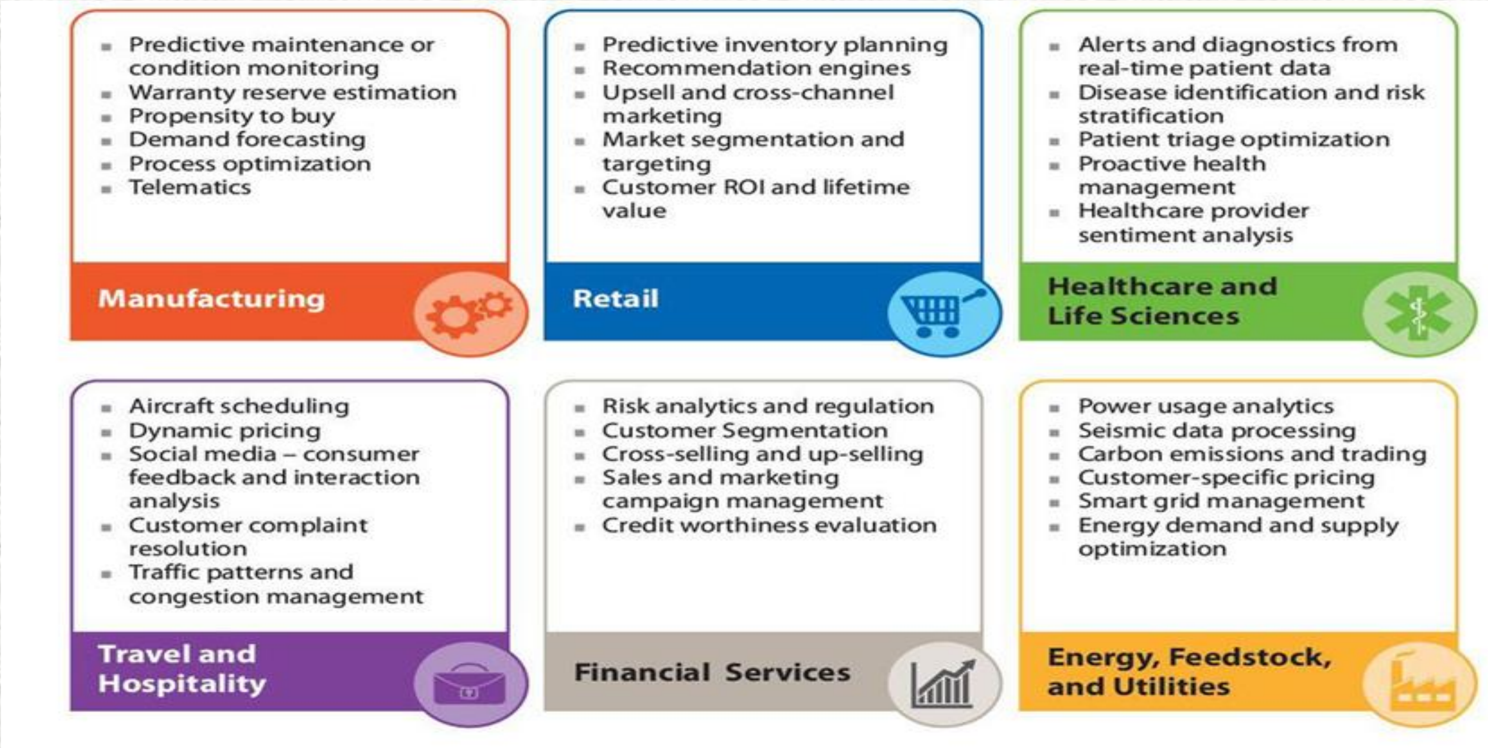
Quick Glance at Deep Learning

Major area of research



Quick Glance at Deep Learning

Applications in various sector



SR-30 Gas Turbine Engine

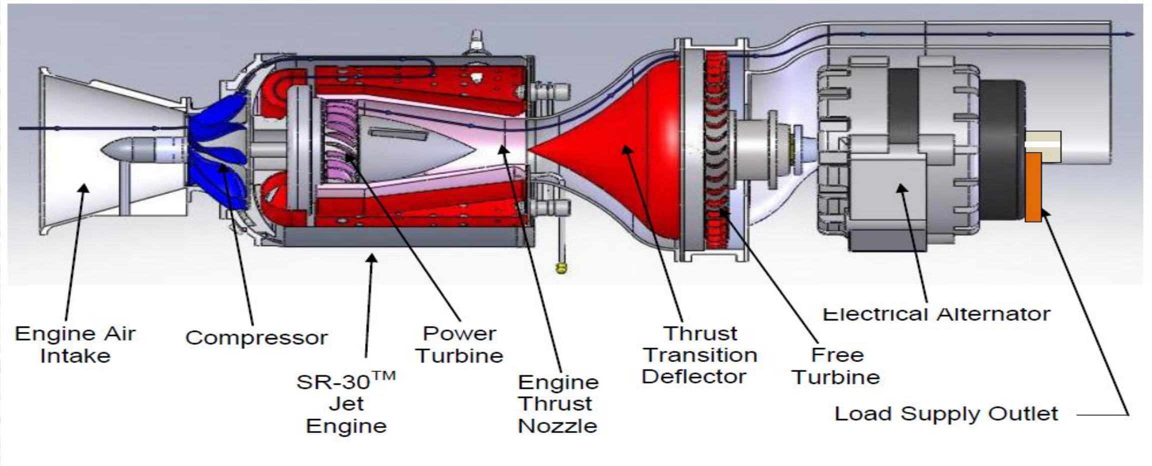


Fig : Cross-sectional view of laboratory SR-30 engine

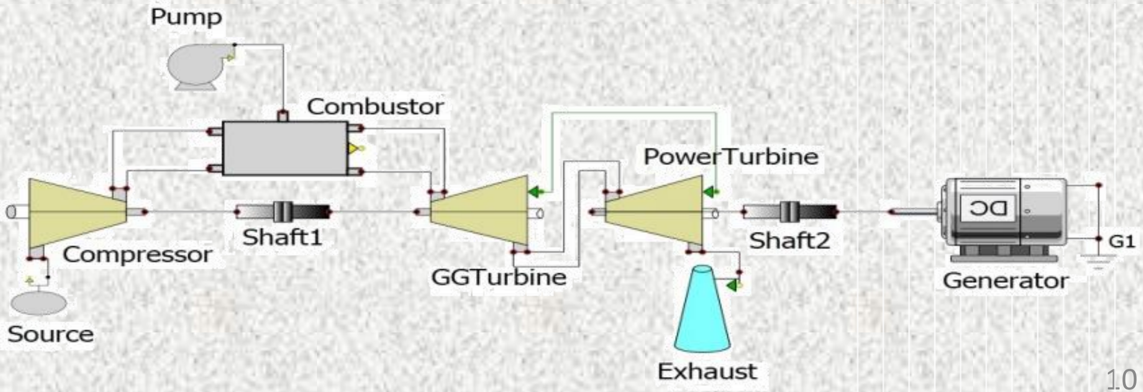


Fig : schematic flow diagram of laboratory engine

Why Engine Model is Necessary??

- To assess real world phenomena
- Ease of dynamic simulation
- Investigating strong adverse dynamic conditions
- Cost saving strategy for performance optimization

Model applications →

- Sensor validation
- Fault diagnosis & detection
- Design and optimization of control system
- Condition monitoring Or engine health monitoring

Dynamic Model of SR-30 Engine

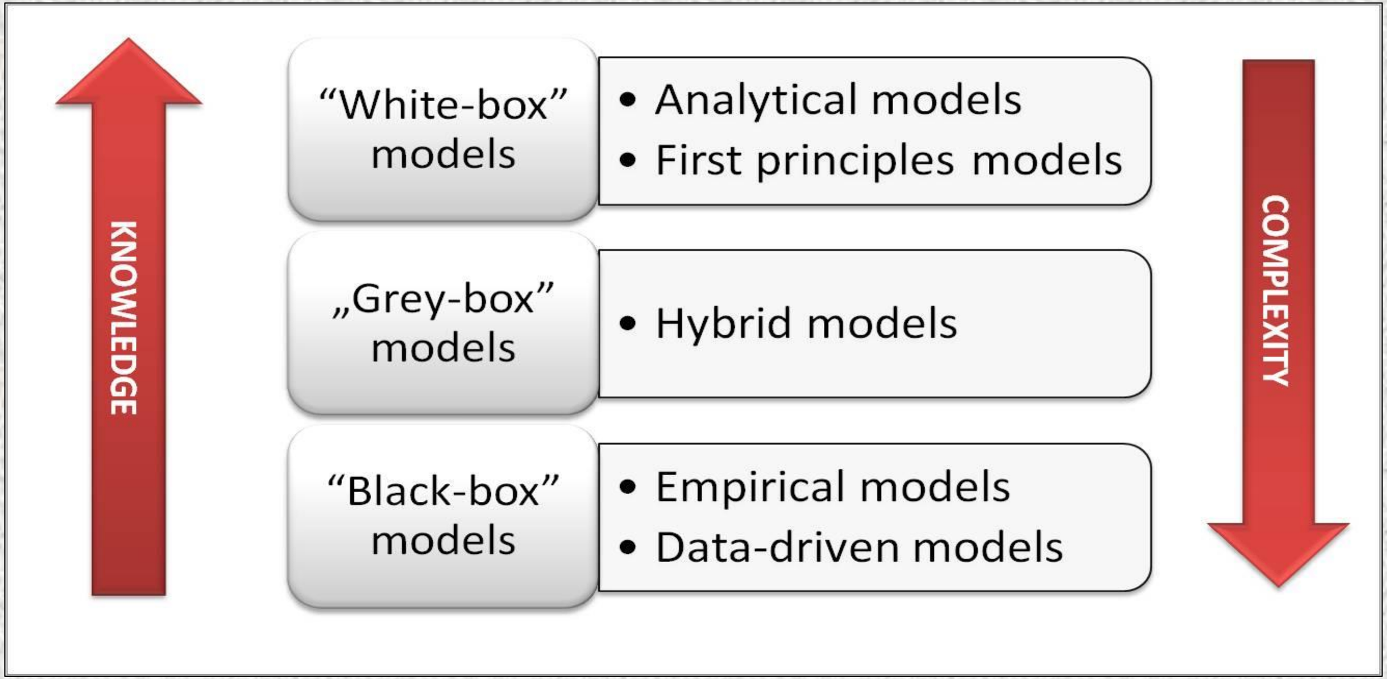
Objective:

- Develop a non-linear dynamic engine model.
- Simulate the steady state and transient performance.
- Integrate the developed gas turbine model with multi-disciplinary systems.

Challenges:

- Experimental data
- Characteristics map of engine components.
- Tuning of characteristics maps.
- Simulate the model over full operating range.

Approaches for Modeling Dynamic Systems



Problem Statement

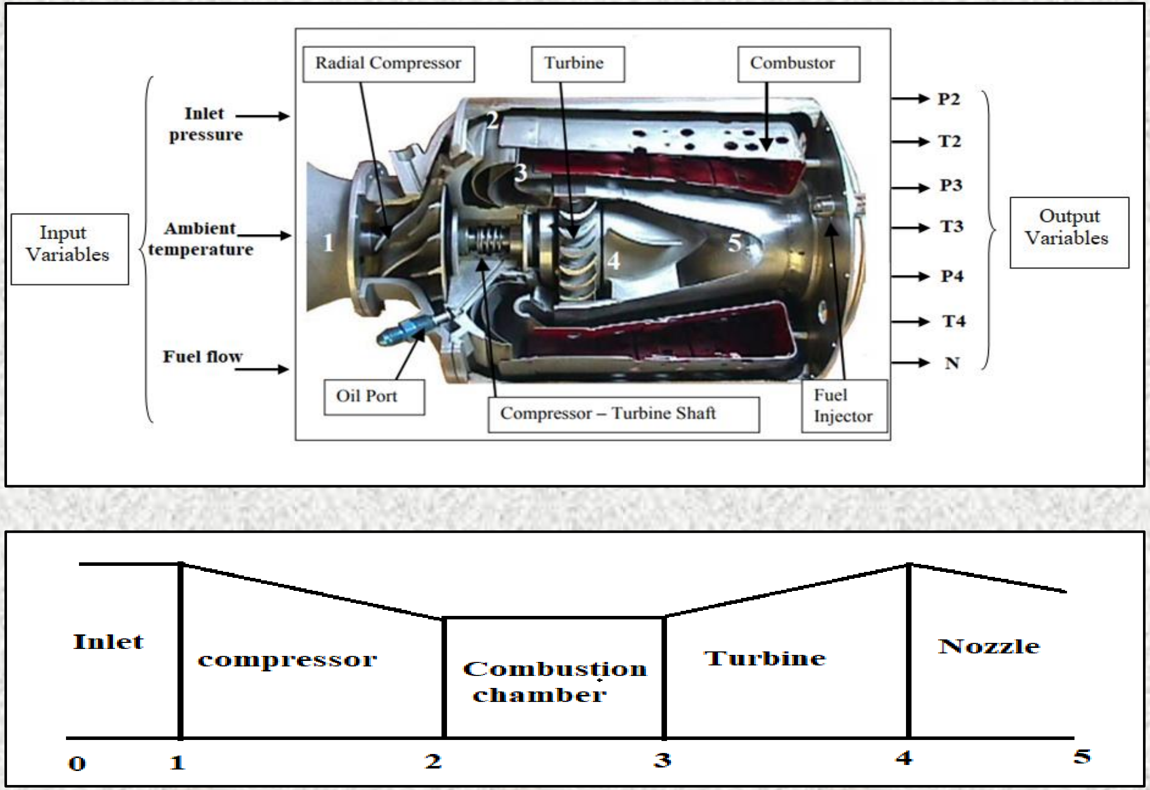


Fig : Illustration of input and output variables of the model

First Principle Modeling Approach

State variable method:

1 - Selection of state Variable

$$x = [P_2 \ \ P_4 \ \ N]$$

2 - Compressor calculation

$$\left(\frac{P_2}{P_1}, \frac{N}{\sqrt{T_1}}\right) \xrightarrow{Compressor\ map} (\dot{m}_c, \eta_c)$$

$$(T_1, \eta_C, PR_{21}) \xrightarrow{isentropic\ equation} (T_2)$$

$$W_c = \dot{m}_c c_p (T_2 - T_1)$$

3 - Combustion chamber

$$(P_2, \sigma_{cc}) = P_3$$

$$(LHV, \dot{m}_c, \dot{m}_f) \xrightarrow{Energy\ balance} T_3$$

4- Turbine calculation

$$\left(\frac{P_3}{P_4}, \frac{N}{\sqrt{T_3}}\right) \xrightarrow{turbine\ map} (\dot{m}_t, \eta_t)$$

$$(T_3, \eta_t, PR_{34}) \xrightarrow{isentropic\ equation} (T_4)$$

$$W_t = \dot{m}_t c_p (T_3 - T_4)$$

5- Nozzle equation

$$\left(\frac{P_4}{P_5}\right) \xrightarrow{nozzle\ map} (\dot{m}_t, \eta_t)$$

First Principle Based Engine Simulator

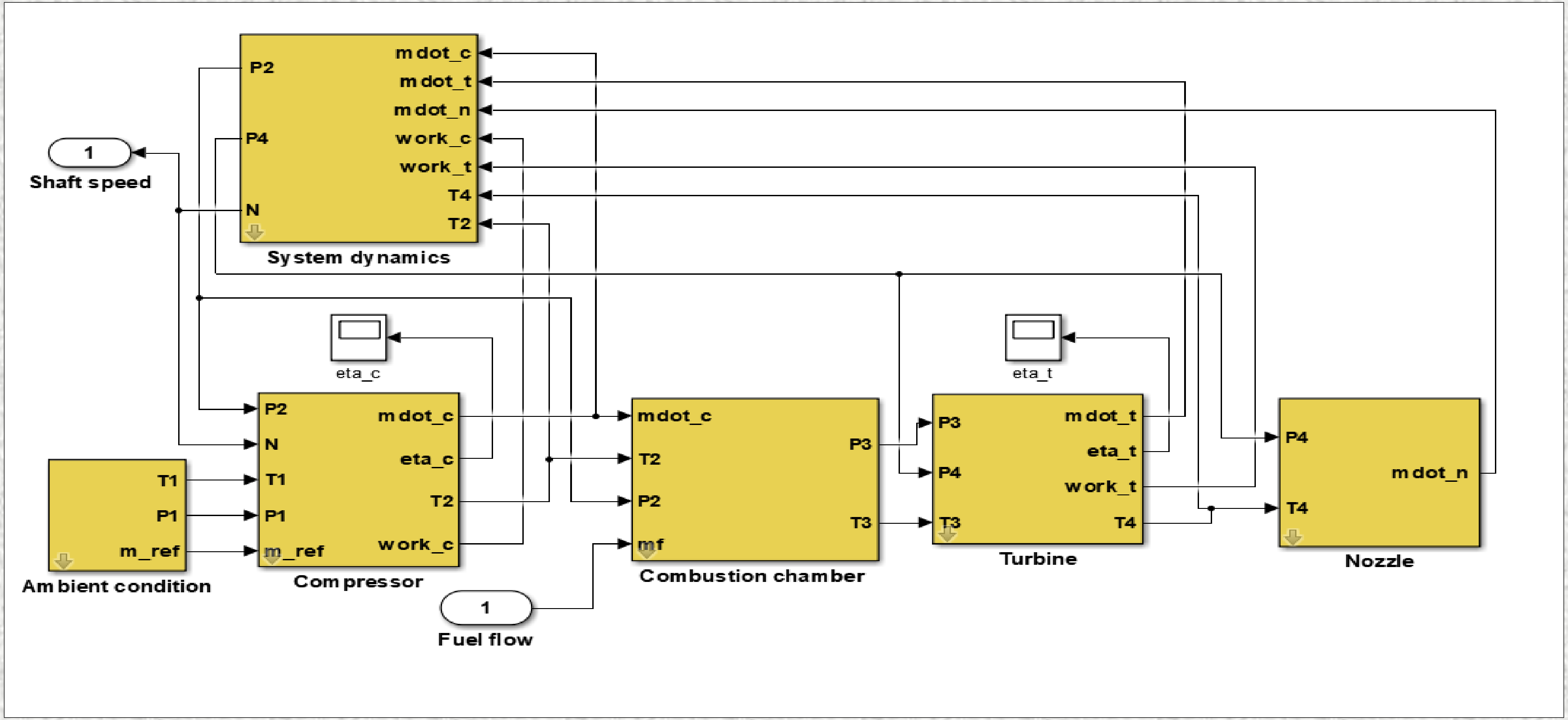


Fig : Component-wise Simulink Model of SR-30 Gas Turbine Engine

First Principle Based Engine Simulator

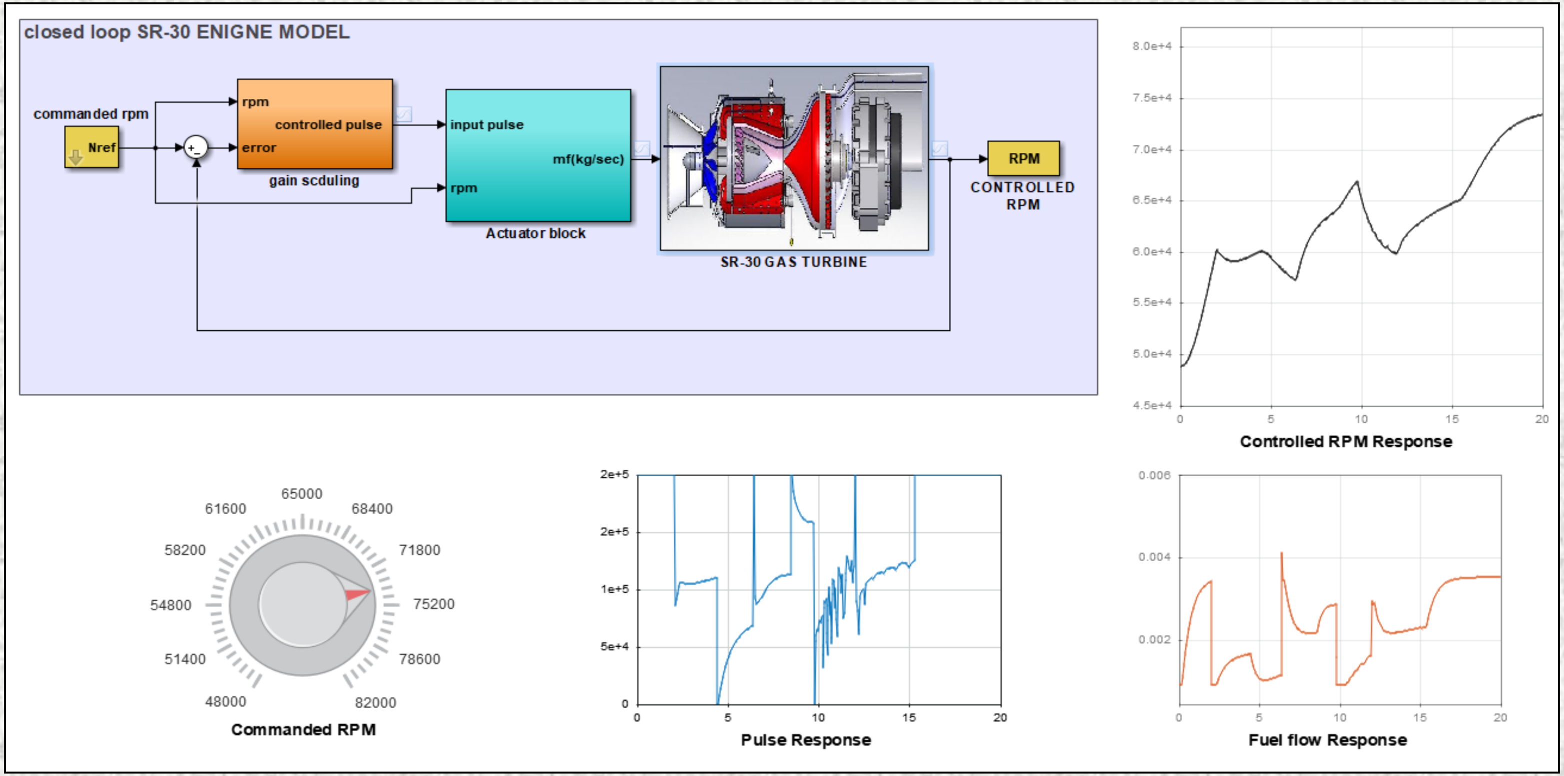


Fig : Closed loop Model of SR-30 Gas Turbine Engine along with dashboard tool

Motivation for Data Driven Techniques

1. White-box or First principles modeling approaches **rely on thermodynamic and energy balance equations**. Hence, assumptions and linearization methods are required to simplify and solve complex dynamics.
2. Models and control systems designed using **simplified linearized equations** are not accurate enough to capture system dynamics precisely.
3. The **unavailability of component maps** is also one of the key reason to shift on data driven modeling techniques.
4. Thus, Deep learning is a fair alternative to white box model as it is independent of the system dynamics with an objective of maximize system robustness, output power and efficiency.

Neural Network Architecture

The model can be mathematically represented as:

$$y(t) = f\,(y(t-1),\ y(t-2), \ldots\ldots,\ y(t-n_y),\ u(t-1),\ u(t-2), \ldots\ldots,\ u(t-n_u))$$

where y (.) is Output, u (.) is Input and n represents the Delay unit.

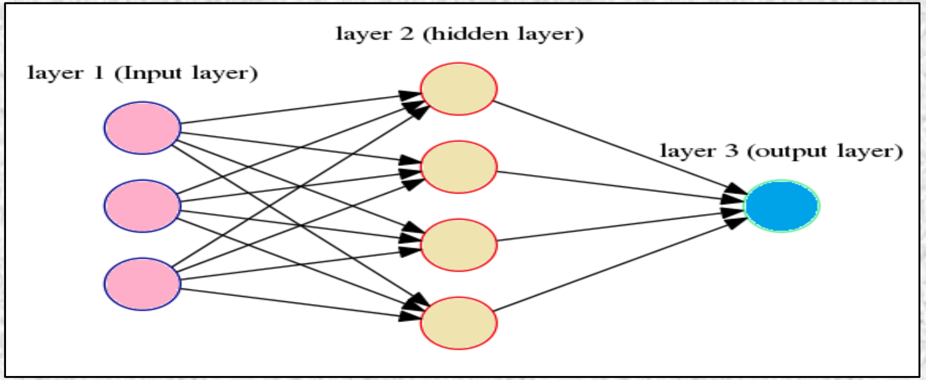


Fig : Network Architecture

Neural Network Architecture

The model can be mathematically represented as:

$$y(t) = f\,(y(t-1),\ y(t-2),\ldots\ldots,\ y(t-n_y),\ u(t-1),\ u(t-2),\ldots\ldots,\ u(t-n_u))$$

where y (.) is Output, u (.) is Input and n represents the Delay unit.

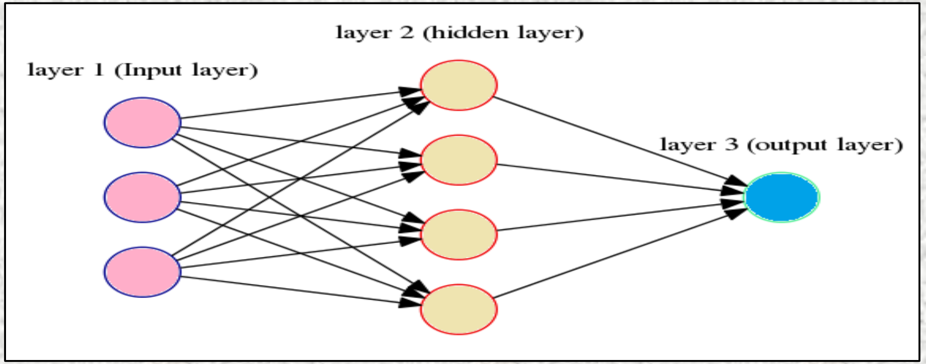
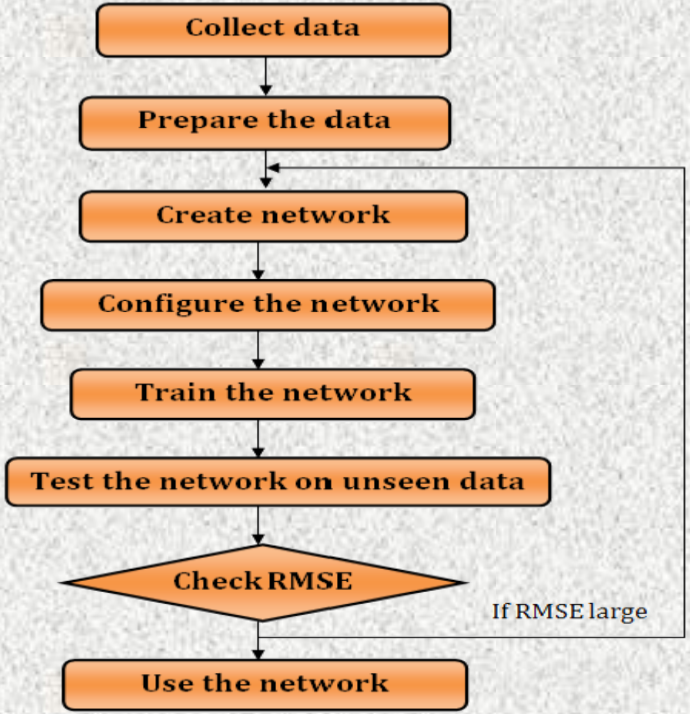


Fig : Network Architecture

How to build Deep neural network?



LSTM Network Architecture

- **Forget gate:**

$$f_t = \sigma(W_f[y_{t-1}, x_t] + b_f)$$

- **Input gate**

$$i_t = \sigma(W_i[y_{t-1}, x_t] + b_i)$$
$$\hat{C}_t = tanh(W_c[y_{t-1}, x_t] + b_c)$$

- **Cell memory state**

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t$$

- **Output gate**

$$o_t = \sigma(W_o[y_{t-1}, x_t] + b_o)$$
$$y_t = o_t * \tanh(C_t)$$

Where: *W* is weight, b is bias, *x* is input data, *y* is target data.

$$tanh(x) = \frac{2}{1+e^{-2x}} - 1 \qquad \sigma(x) = \frac{1}{1+e^{-x}}$$

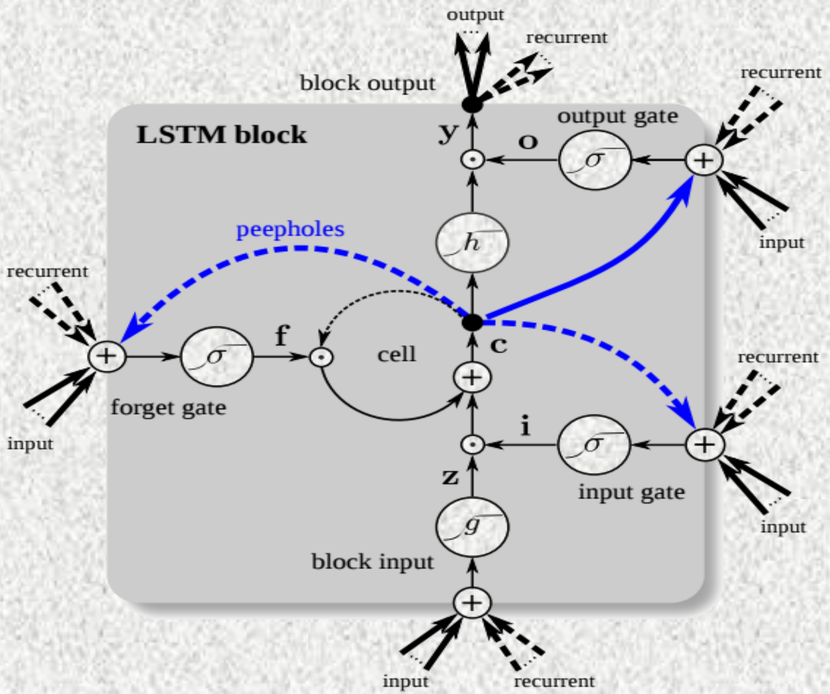
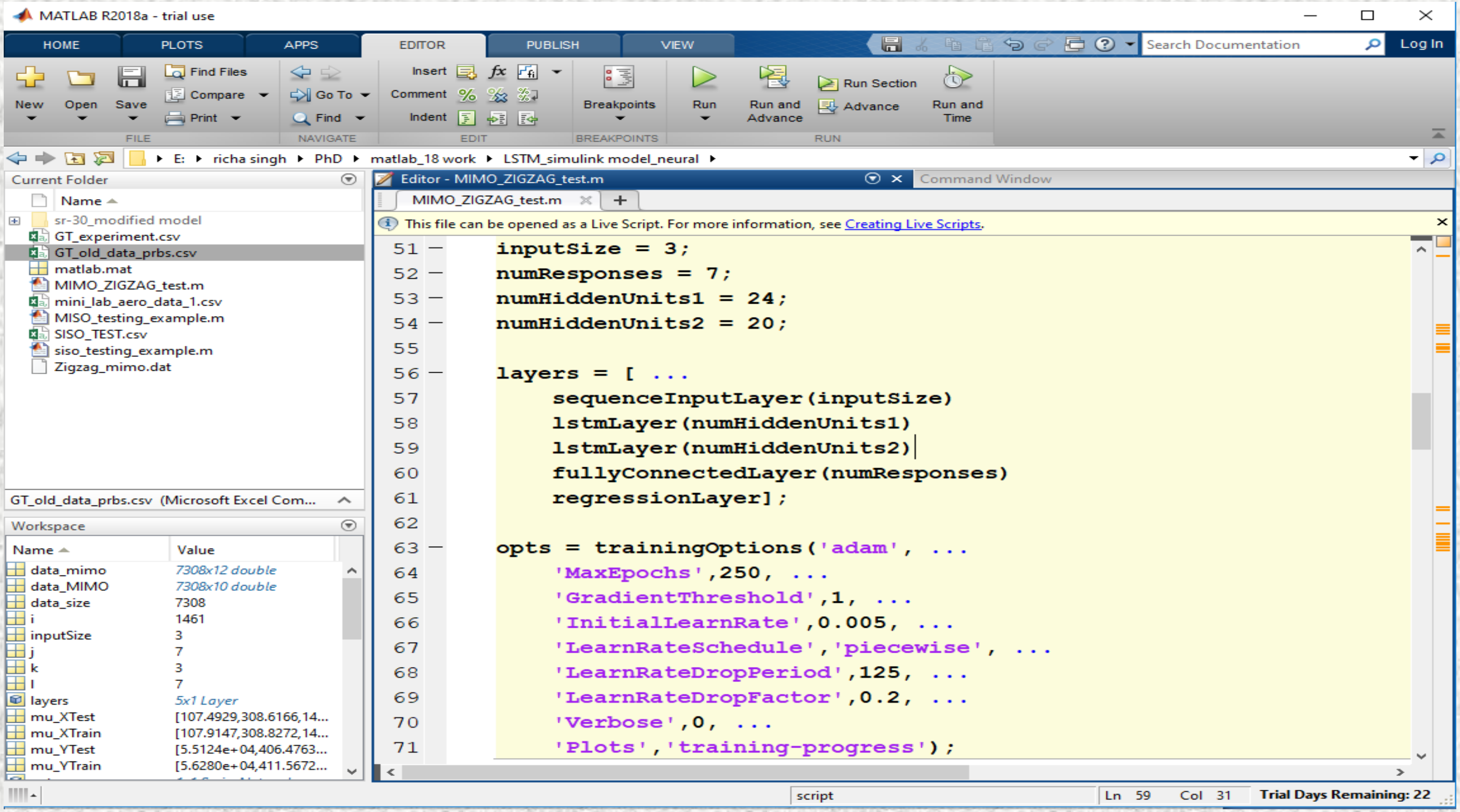


Fig : Detailed schematic of LSTM block

Network Configuration



Model Validation Against Experimental Data

The validation results of First principle model as well as Deep learning based model against experimental data is represented for shaft speed (full range RPM)

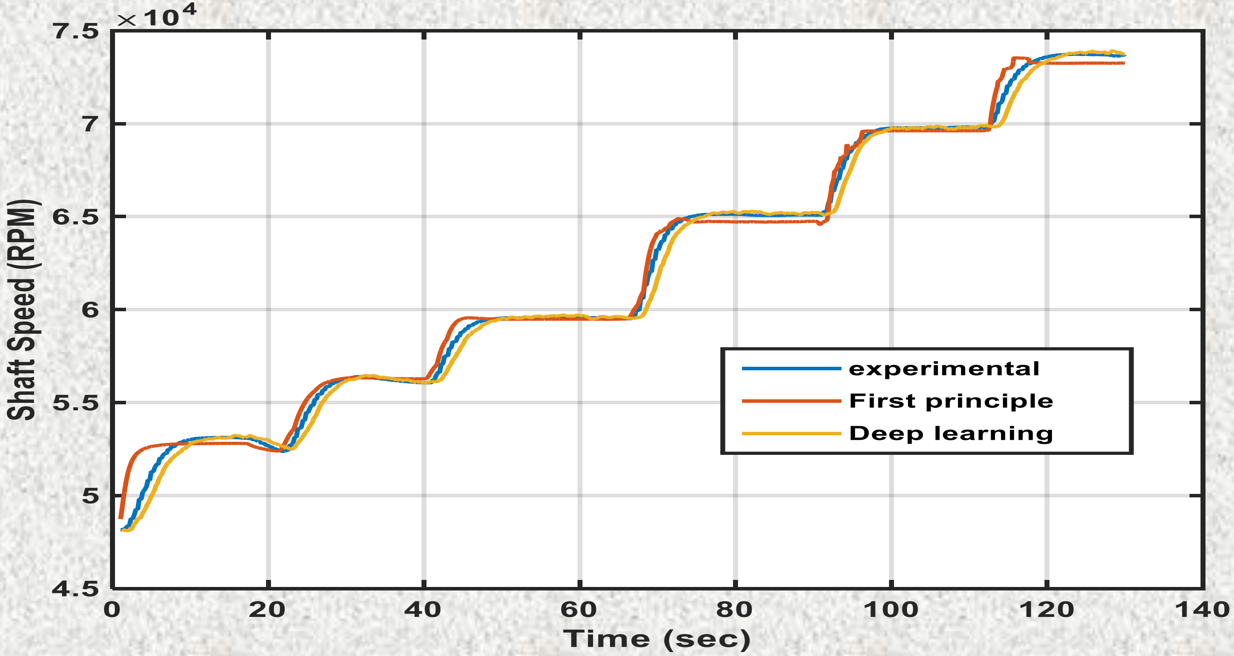
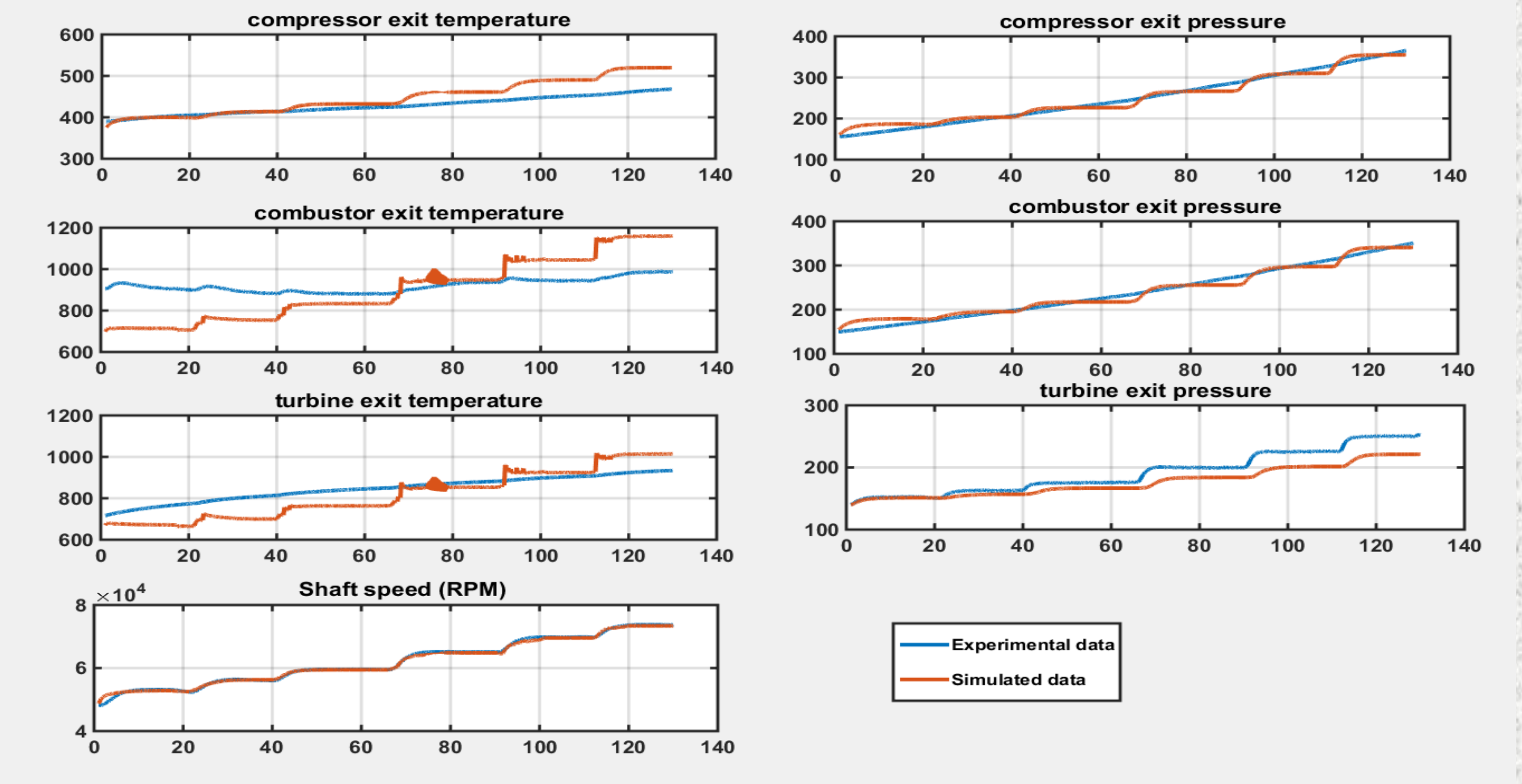
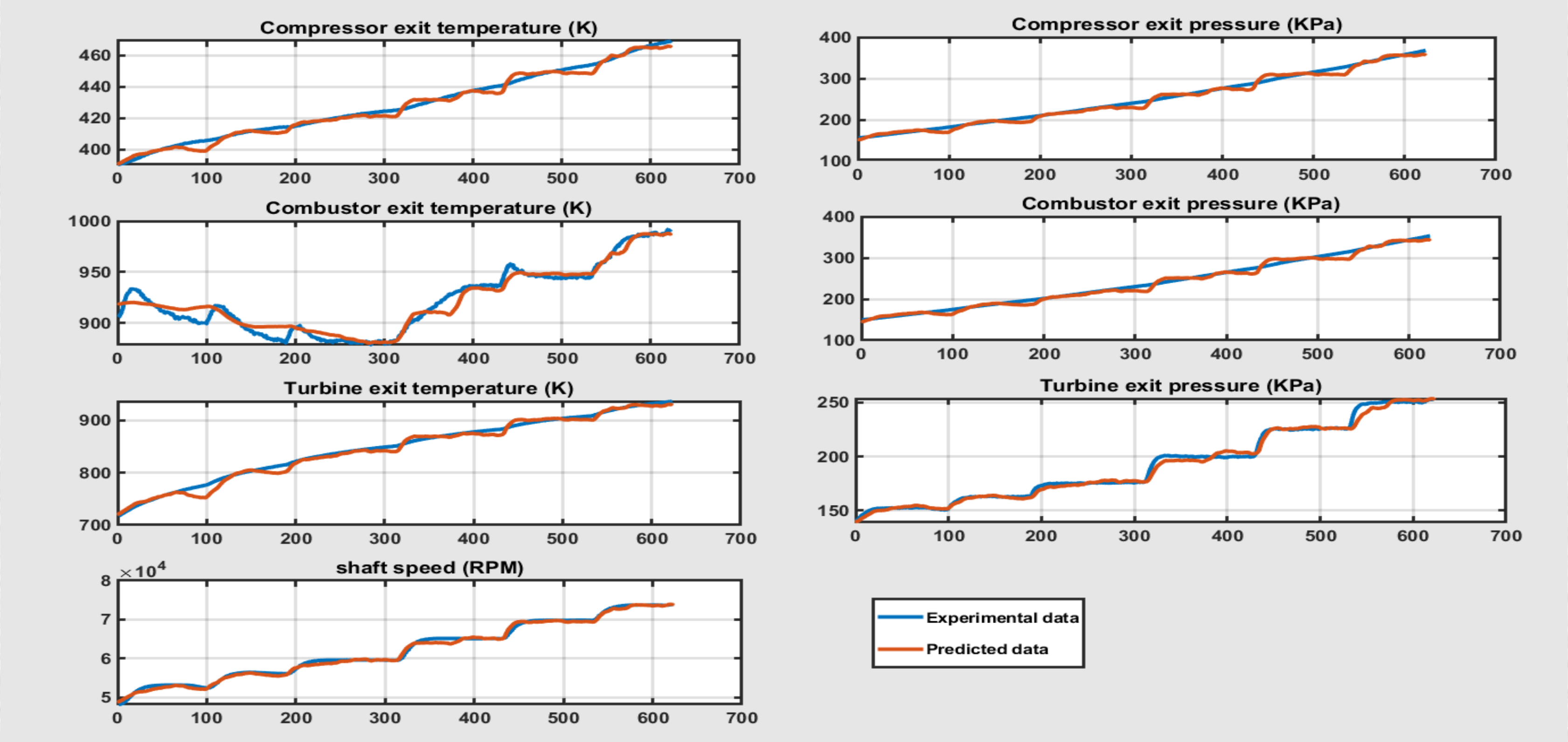


Fig: Shaft speed validation using First Principle model and Deep Learning model

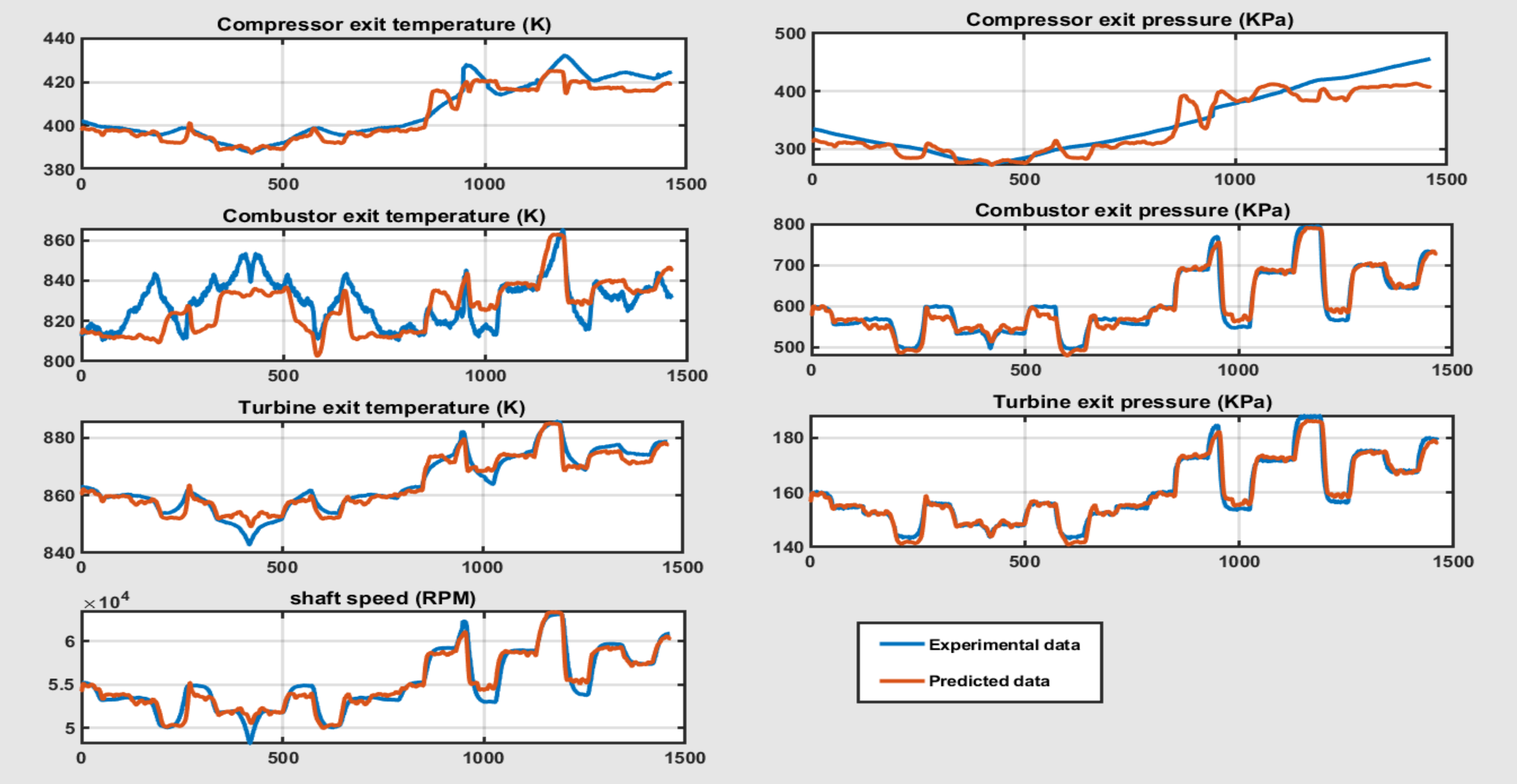
First Principle Model Validation Against Experimental Data



Deep Learning Model Validation Against Experimental Data



Deep Learning Model Validation Against Experimental Data



Relative Error of predicted data against experimental data

$$Error = \frac{1}{N}\sum\left|\frac{y_{exp} - y_{pred}}{y_{exp}}\right|$$

Parameter	Deep Learning Approach	First Principle Approach
T2	0.002	0.0441
P2	0.0012	0.0112
T3	0.0031	0.1297
P3	0.0011	0.0255
T4	0.009	0.1428
P4	0.0036	0.0721
N	0.001	0.0014

Conclusions

- First Principle based method promises good dynamic behavior when compared with the real time engine, provided that enough information is available.

- The deep learning model is trained with a set of experimental data which makes the model to learn a wide variety of engine behavior.

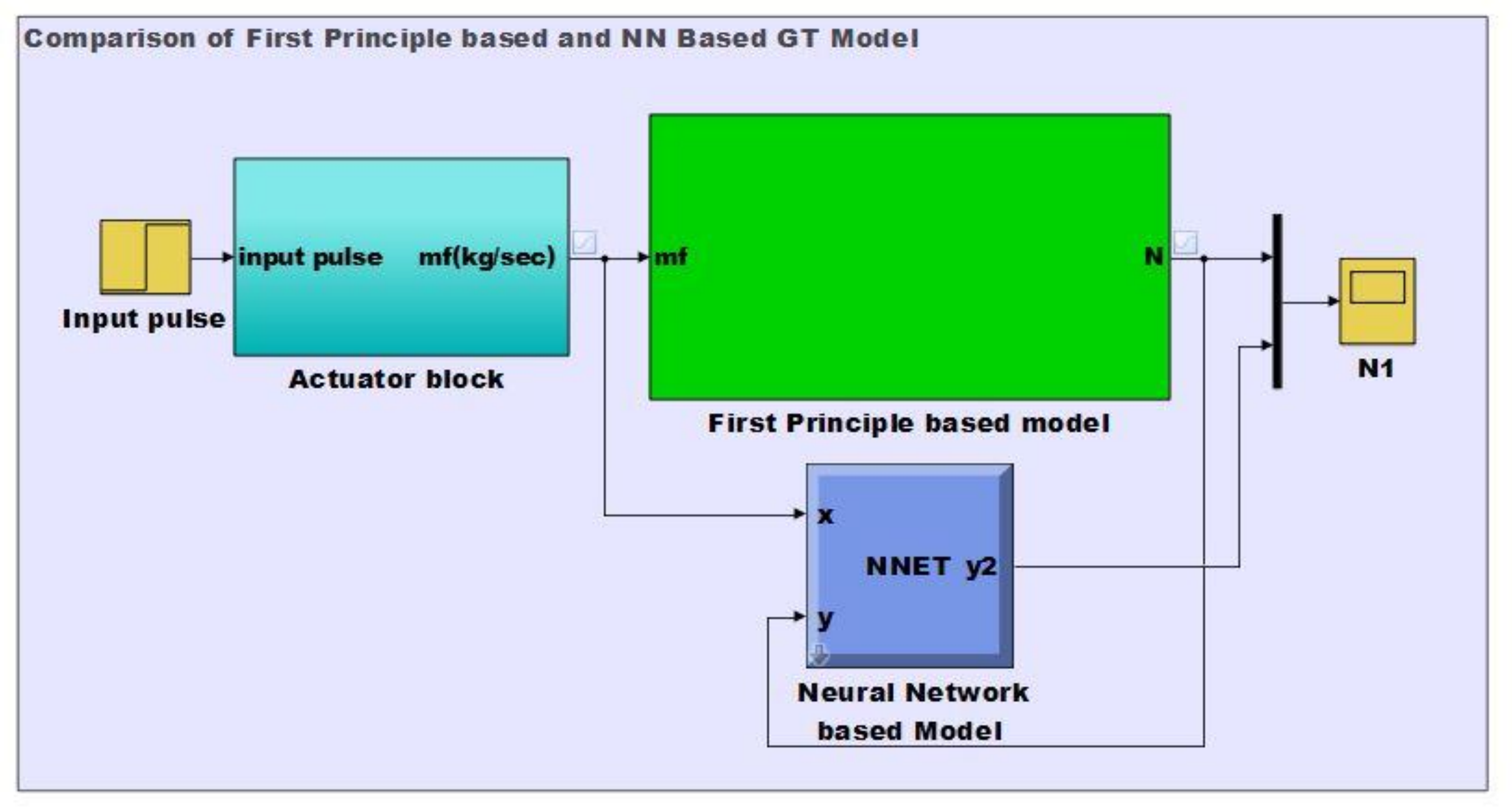
- The **Deep Learning** approach when compared with the First Principle model against experimental data is found to be **more efficient** in predicting behavior of system.

- LSTM performs **good with MIMO system**, however LSTM has its own disadvantage: it is slower than other normal activation functions which leads to the trial & testing process to be more slow.

Why Matlab & Simulink?

- Easy user-interface
- Less programming required while working in Simulink.
- Toolboxes are designed to integrate with parallel computing environments, GPUs, and automatic C code generation.
- Documentation is written for engineers and scientists, not computer scientists.
- Inbuilt functions are available that are required in day-to-day computation.
- MATLAB App let you start working right away and then automatically generate a MATLAB program to reproduce or automate your work.

Future Steps in Modeling and Control of Engine



Acknowledgements

Experimental setup : Turbine Technologies, LTD.

Creating Innovative Laboratory Systems For Engineering & Technical Education

- **Supervisor:**
  - P. S. V. Nataraj
- **Co - worker**
  - Bhagyashri Somani (deep learning)
- **Data collection from experimental setup:**
  - Swathi Surendran
  - Sanjeet Kulkarni


Thank You