Background

Related work

Methodology

Architecture

Optimization

Results

Conclusion

Architecture

Evaluation

Evaluating Machine Learning Models for Minimizing Downtime through Operator Task Scheduling in CNC Production Lines

Student: Kaan Özsan
Mail: ozsan@kth.se
Organization: Scania

Date: 03-06-2025

Supervisor: Yasaman Khorsandmanesh
Examiner: Håkan Olsson

Background

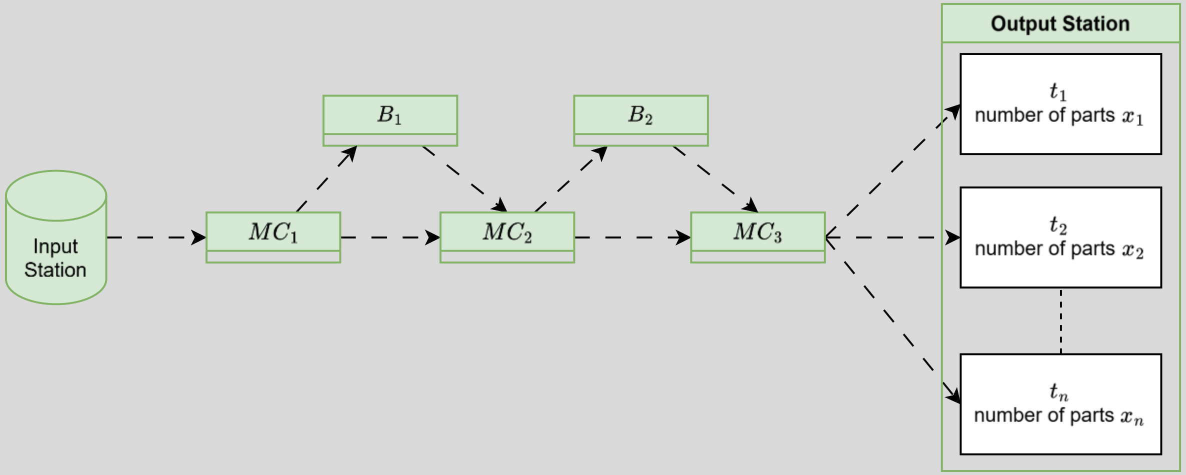
Research Question

Which machine learning models perform best in predicting and scheduling operator tasks in a CNC production line environment, based on metrics such as accuracy, precision, latency, and robustness?

Sub-questions include:

- How do different models compare in handling real-time or simulated data for predictive task planning?
- Which model performs best at classifying operator task occurrences relevant to operator scheduling?
- Which machine learning approach, in the context of continuous learning, results in the least downtime during model updates and retraining?

Background



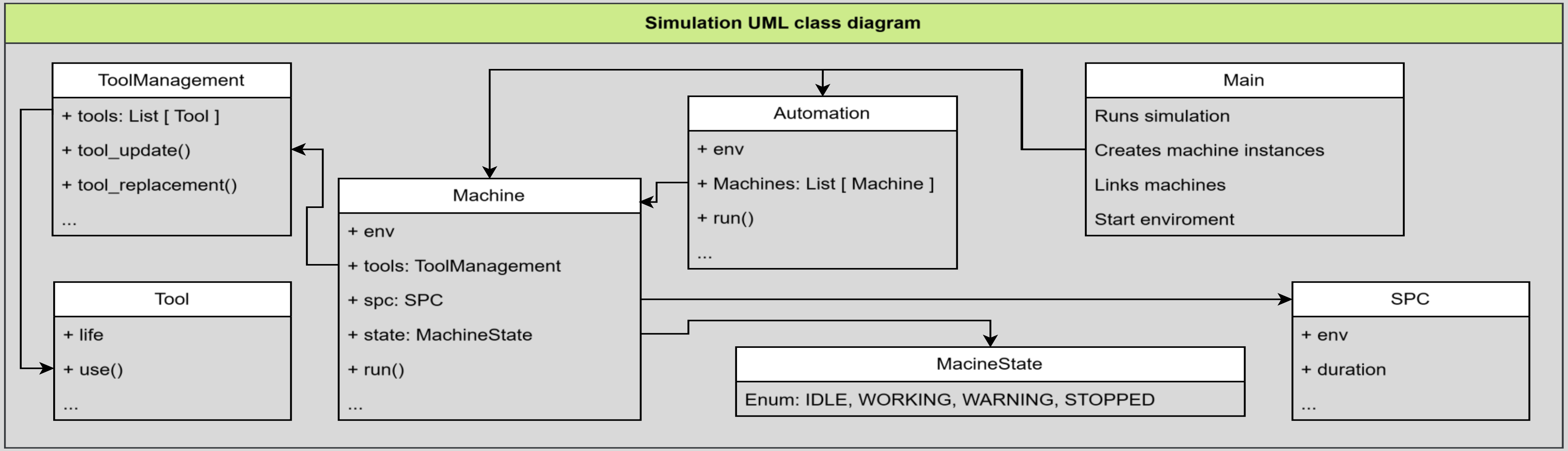
- CNC lines produce parts using machines that require periodic operator-performed maintenance.
- Common tasks include **tool changes** and **quality checks**, which may affect line throughput.
- Task timing may shift if machine dependencies cause production to pause

  (e.g., due to upstream/downstream stoppages).
- Overlapping tasks (e.g., multiple tool changes at once) increase operator load and response time.

Related work

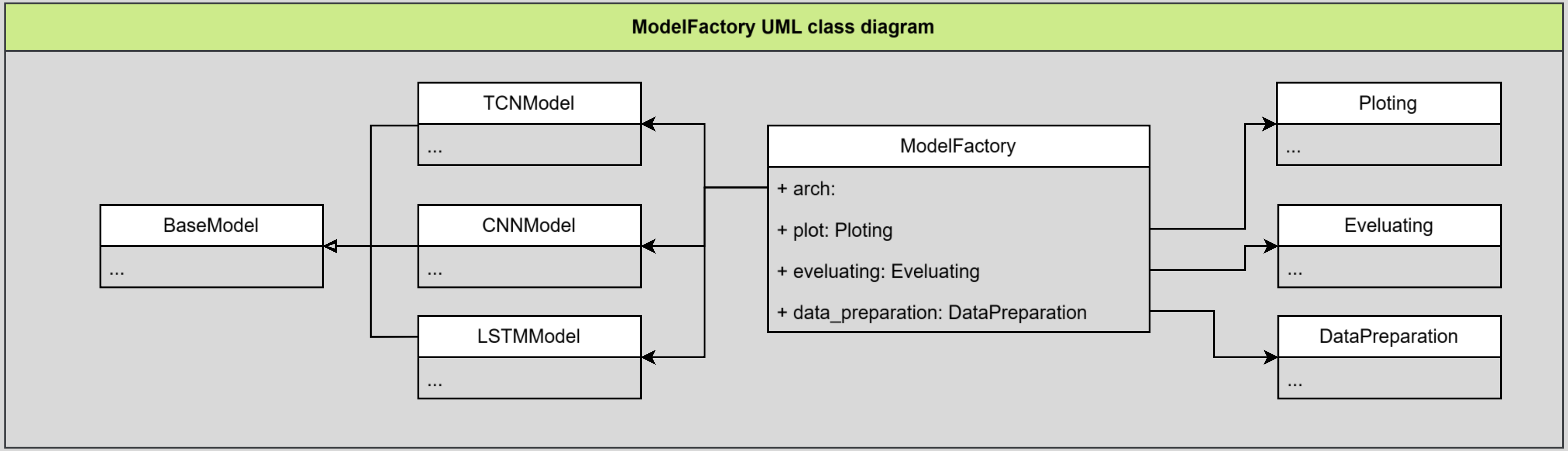
- No prior studies target operator task prediction in CNC environments.
- Smart maintenance research applies ML to forecast tool wear and failures.
- Techniques include time-series models and event-driven forecasting.
- This thesis adapts those methods to model operator-dependent interventions.

Methodology



- A simulation program was developed to generate machine signal data based on Scania's current signal structure and historical patterns.
- The simulation replicates only the tracked signals, limiting completeness but ensuring realistic data shape.
- Since the data is synthetic and structured, no imputation or normalization was required in preprocessing.
- The generated sequences were transformed into time-series inputs for machine learning models.

Architecture



TCN (Temporal Convolutional Network)

- Captures long-range temporal patterns using dilated causal convolutions.
- Well-suited for fixed-length sequences and multi-label forecasting.

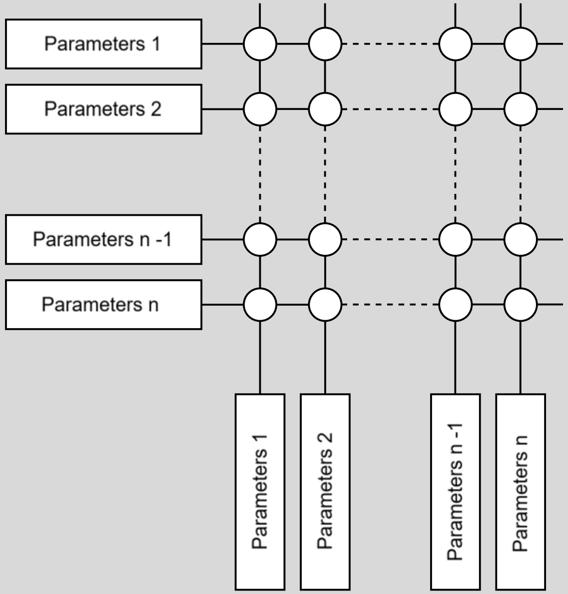
CNN (Convolutional Neural Network)

- Learns spatial patterns across time steps using convolution filters.
- Effective at capturing localized signal features like sudden changes or cycles.

LSTM (Long Short-Term Memory)

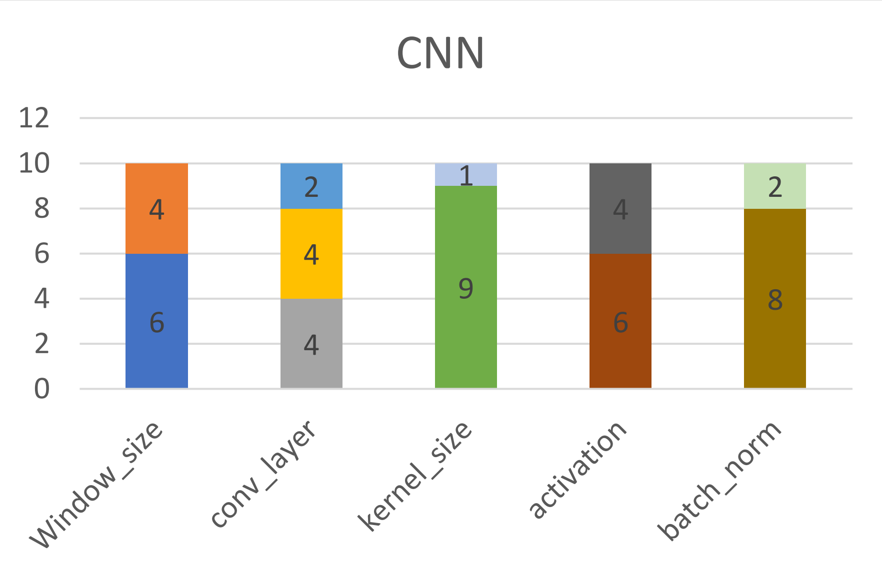
- Designed to learn long-term dependencies in sequential data.
- Handles variable-length input and internal state over time steps.

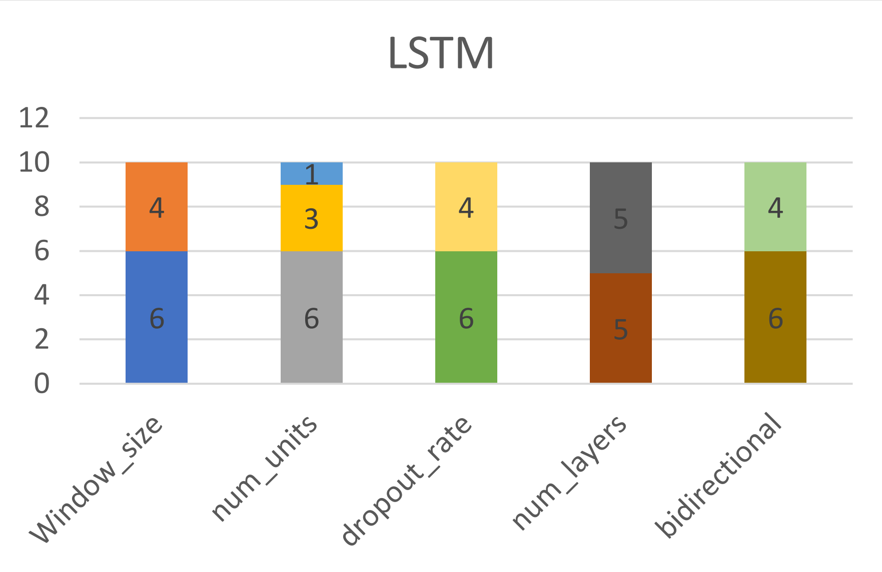
Optimization

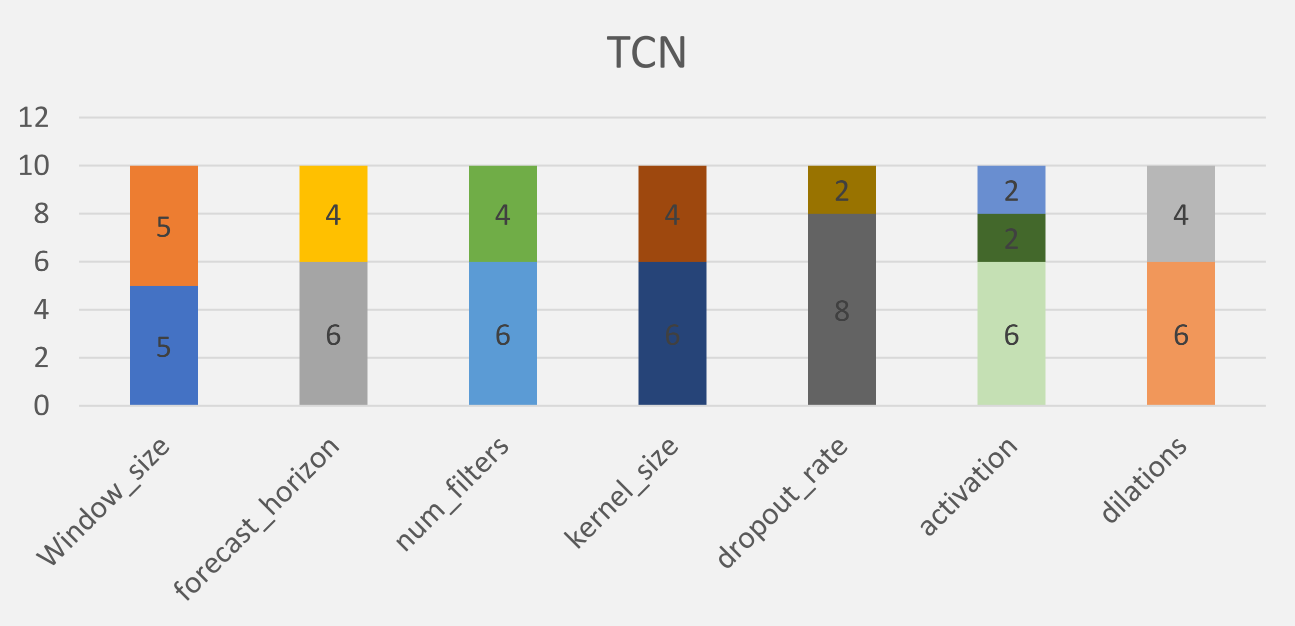


- Grid search was used to explore different hyperparameter configurations for each model.
- All models were trained using the same 3-month training set and validated on the same validations data.
- Due to architectural differences, each model required its own tailored search space
- While computationally expensive, this approach revealed both optimal settings and deeper insights into input data characteristics.

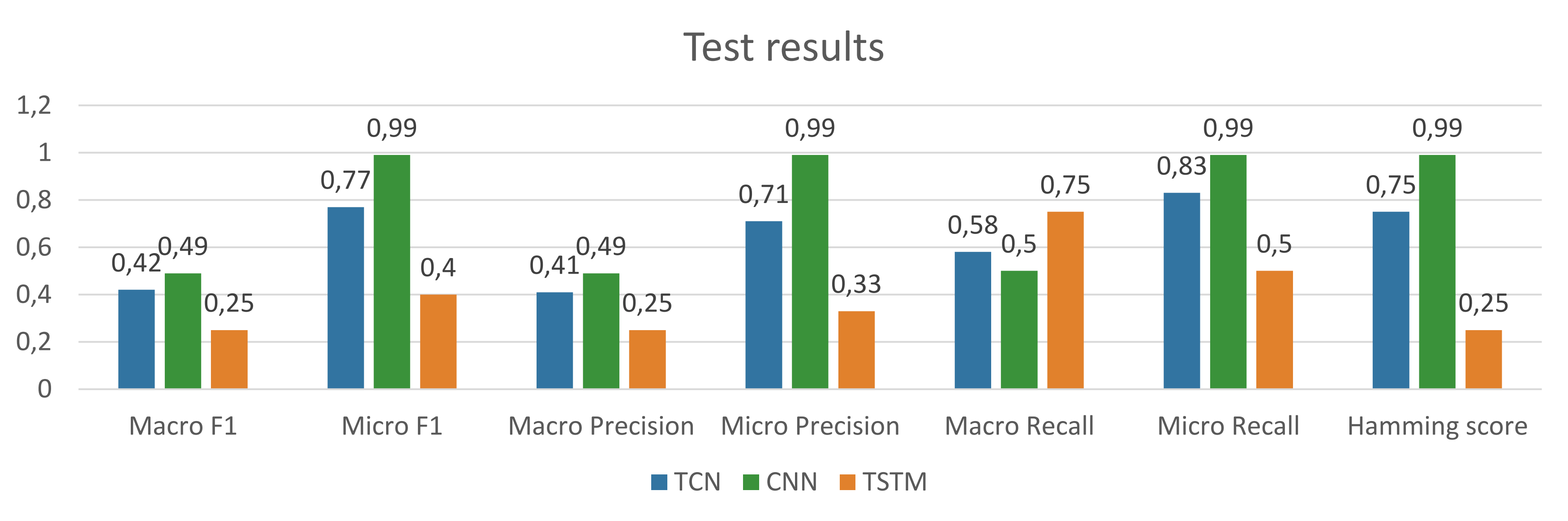
Optimization



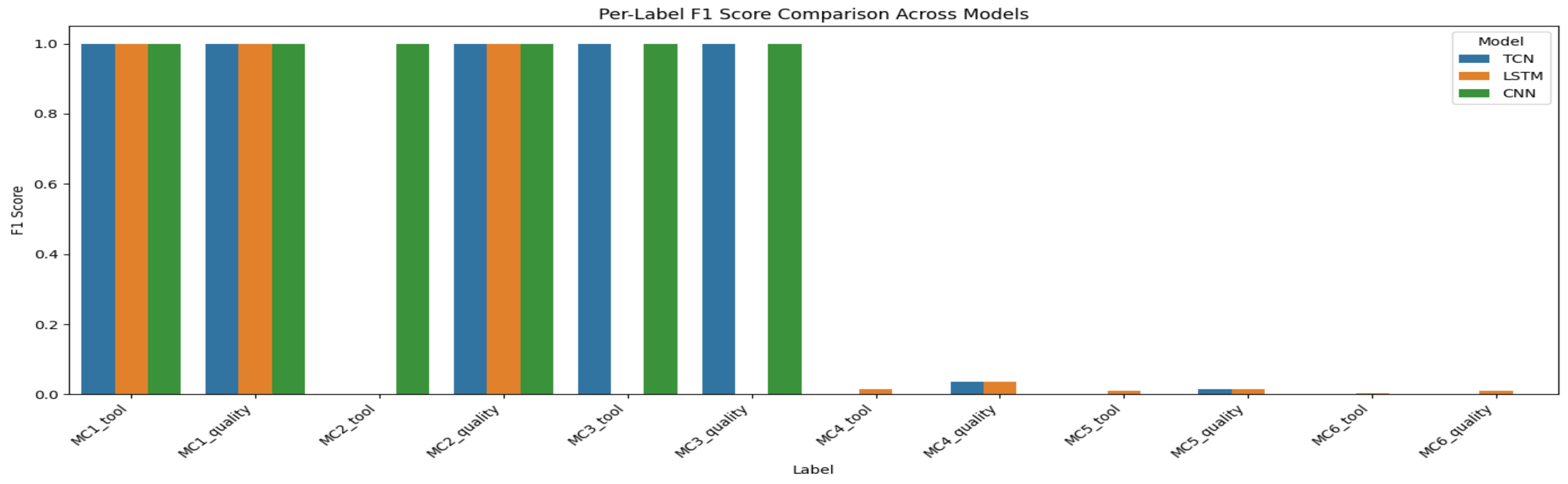




Results



Test results
1,2
1
0,8
0,6
0,4
0,2
0
0,42
0,49
0,25
0,77
0,99
0,4
0,41
0,49
0,25
0,71
0,99
0,33
0,58
0,5
0,75
0,83
0,99
0,5
0,75
0,99
0,25
Macro F1
Micro F1
Macro Precision
Micro Precision
Macro Recall
Micro Recall
Hamming score
TCN
CNN
TSTM



Per-Label F1 Score Comparison Across Models
F1 Score
1.0
0.8
0.6
0.4
0.2
0.0
Model
TCN
LSTM
CNN
MC1_tool
MC1_quality
MC2_tool
MC2_quality
MC3_tool
MC3_quality
MC4_tool
MC4_quality
MC5_tool
MC5_quality
MC6_tool
MC6_quality
Label

Conclusion

- The results do not clearly indicate a single best-performing model across all machines.
- For Machine 1, F1 scores were similar across models, but performance diverged further along the line.
- As task frequency decreased due to internal dependencies, model performance also declined, particularly for downstream machines.
- The last machine showed weak results for all models, with only LSTM showing signs of learning.

Future Work

- Extend from a single-model to a **multi-model system**, with specialized models per task (e.g., tool change, quality check).
- Explore **machine-specific models**, each tailored and optimized for the behavior of its corresponding CNC machine.
- Use **Graph Neural Networks (GNNs)** to model task dependencies across machines in the production line.
- Investigate **Reinforcement Learning (RL)** to dynamically schedule operator tasks using throughput-based reward signals.

Thank you for your attention.

I look forward to your questions and feedback.