Combining patient visual timelines using deep learning to predict mortality

Matthew Lind | lind6
Zachary Moulton | moulton6

CS 598 Deep Learning for Healthcare
University of Illinois at Urbana-Champaign

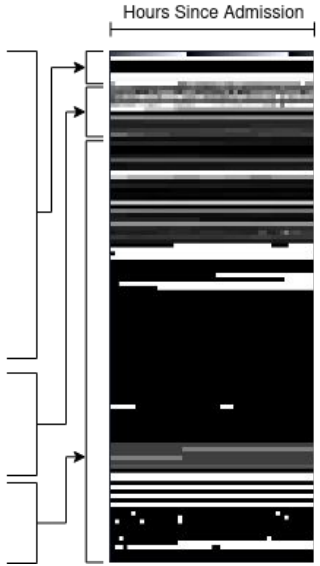
General Problem

- Objective:
  - Predicting in-hospital mortality
- Problem:
  - No direct mortality prediction tools exist
  - Existing tools predict need of immediate medical attention, not mortality
- Proposed solution:
  - Deep-learning based mortality prediction tool

Specific Approach

- Convert patient data to visual timelines.
- Choose key variables relevant to mortality
- Normalize according to ‘custom’ or ‘min/max’ schemes

Patient Characteristics	Demographics: Age, Sex, Race	Continuous - age Binary – sex (male vs. female), race (black vs. other)
	Prior conditions: Prior cardiac arrest, Prior admission within 90 days	Binary
Time	Hour of day	Continuous
Location	Current location (ICU vs. ED vs. other)	Categorical
Vital signs	Temperature (C°), Heart Rate, Respiratory Rate, Systolic Blood Pressure (SBP), Diastolic Blood Pressure (DBP), O2 Saturation, FiO2, AVPU	Continuous
Labs, medications, etc ...	...	...



Specific Approach

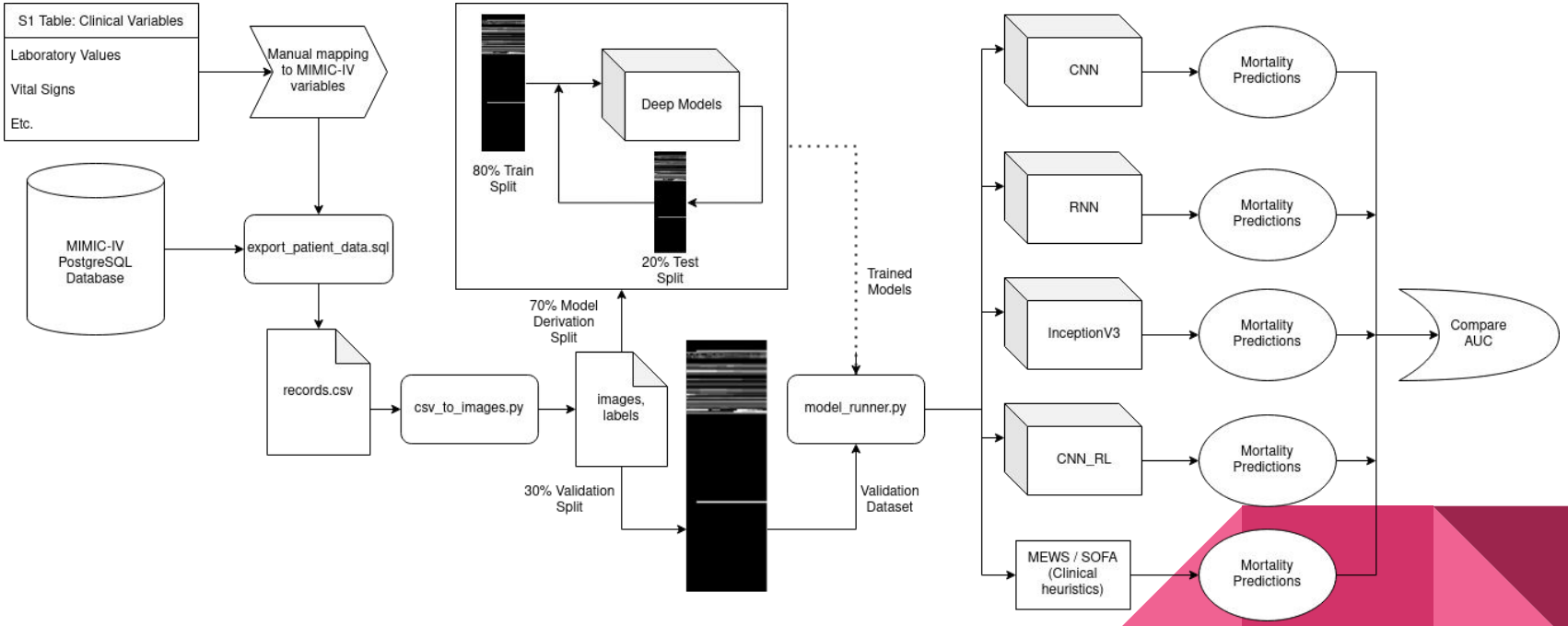
- CNN with recurrent layer
- Outperforms baseline deep learning models
- Combines advantages of CNN and RNN
  - CNN for image prediction
  - RNN for time series prediction

Table 2. Model discrimination for predicting mortality on the test dataset (n = 34,747 admissions).

Model	AUC (95% CI)	*P*-value (compared to CNN-RL)
SOFA	0.57 (0.55, 0.59)	<0.001
MEWS	0.76 (0.74, 0.78)	<0.001
Standard-CNN	0.87 (0.85, 0.88)	<0.001
RNN	0.89 (0.88, 0.91)	0.003
Deep-CNN	0.90 (0.89, 0.91)	0.025
CNN-RL	0.91 (0.90, 0.92)	-

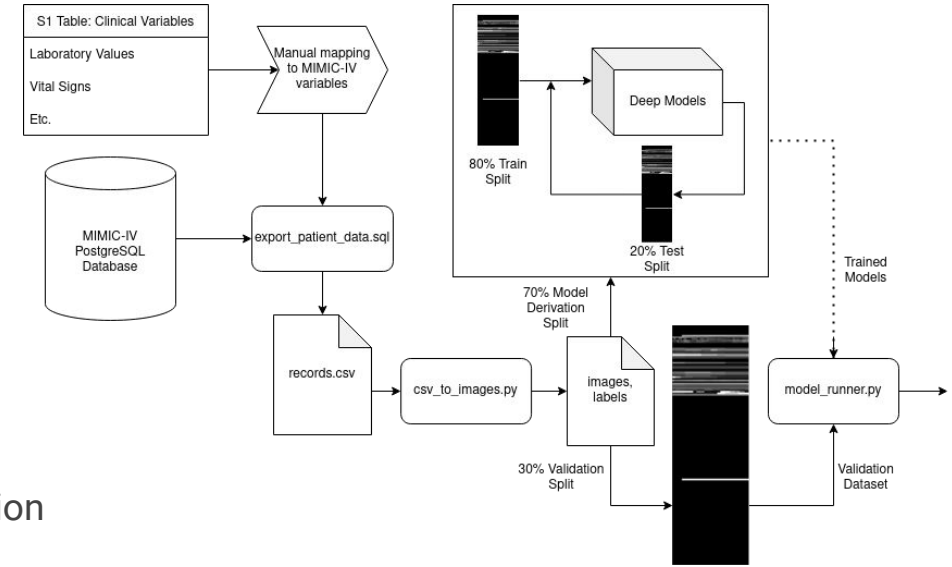
SOFA: Sequential Organ Failure Assessment score
MEWS: Modified Early Warning Score
CNN: Convolutional Neural Network
RNN: Recurrent Neural Network
RL: Recurrent Layer
AUC: Area Under the receiver operating characteristic Curve
CI: Confidence Interval

Reproduction Steps



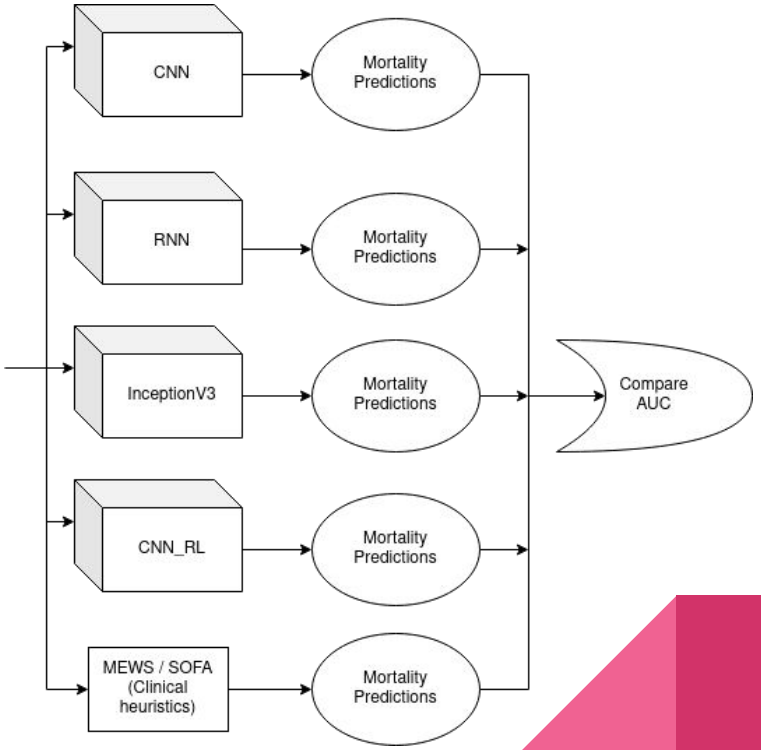
Reproduction Steps: Data Aggregation

- SQL: extract MIMIC data to CSV
  - Parse CSV with Python
    - Generate images
    - Compute MEWS, SOFA
    - Build cohort
      - Shuffle randomly
      - Split into train, test, validation



Reproduction Steps: Deep Learning

- Learning
  - Train and evaluate all models
  - Multiple runs
  - Validate on different cohorts
- Compare results



Reproduction: Results

- Framework was reproducible
  - AUCs imply reasonable predictions
- Default hyperparameters need tuning

n = 337959			
Model	**AUC**	**95% CI-**	**95% CI+**
SOFA	0.67		
MEWS	0.68		
Standard-CNN	0.87	0.86	0.88
RNN	0.88	0.88	0.89
Deep-CNN	0.90		
CNN-RL	0.75	0.57	0.93