

Experience Replay



THE UNIVERSITY OF
SOUTHERN MISSISSIPPI®

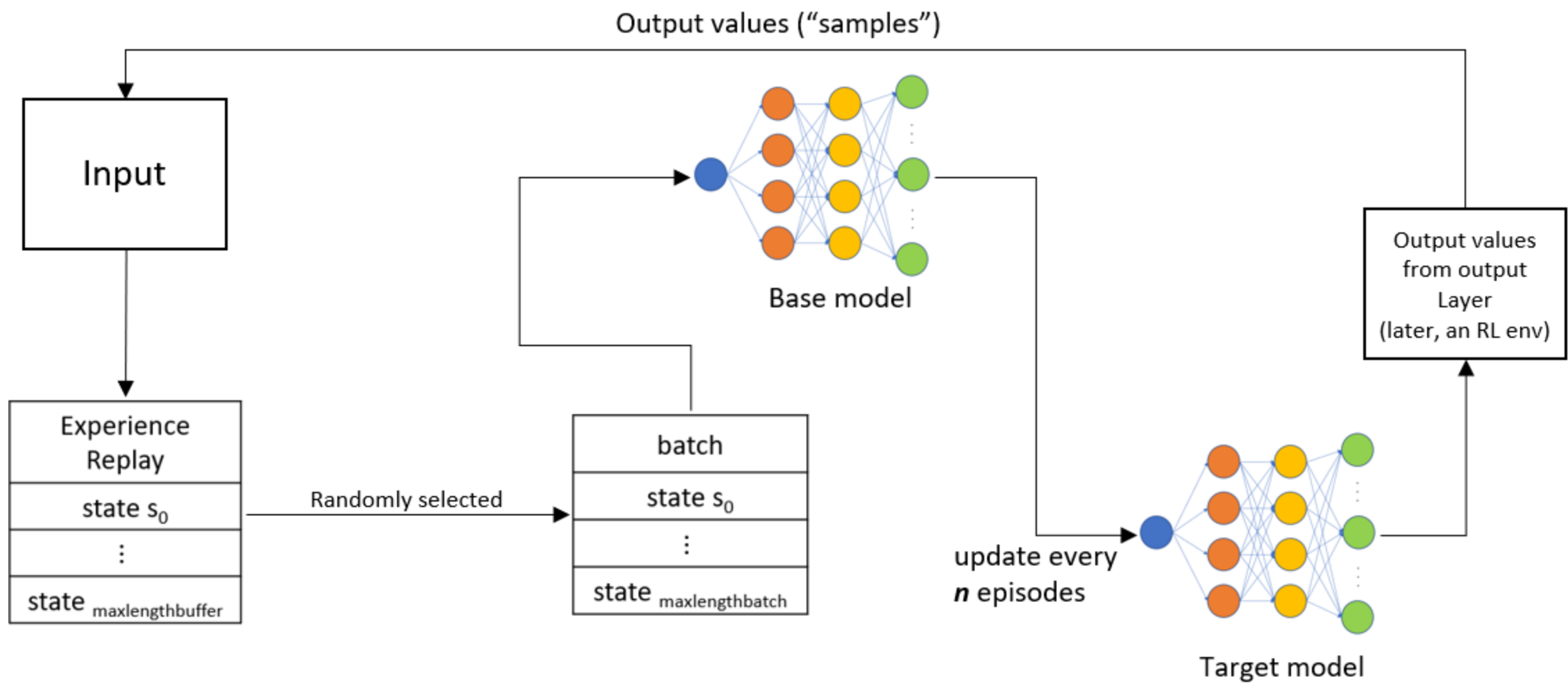
Experience Replay (ER)

- Implemented to increase the efficiency of a training model
 - Samples are randomly selected and saved at each step during an episode
 - Samples are randomly selected to train the model
 - Samples come from the output layer of the NN
 - This will change to output of an RL environment
- ER
 - Smoothens fluctuations
 - Reduces correlation of samples

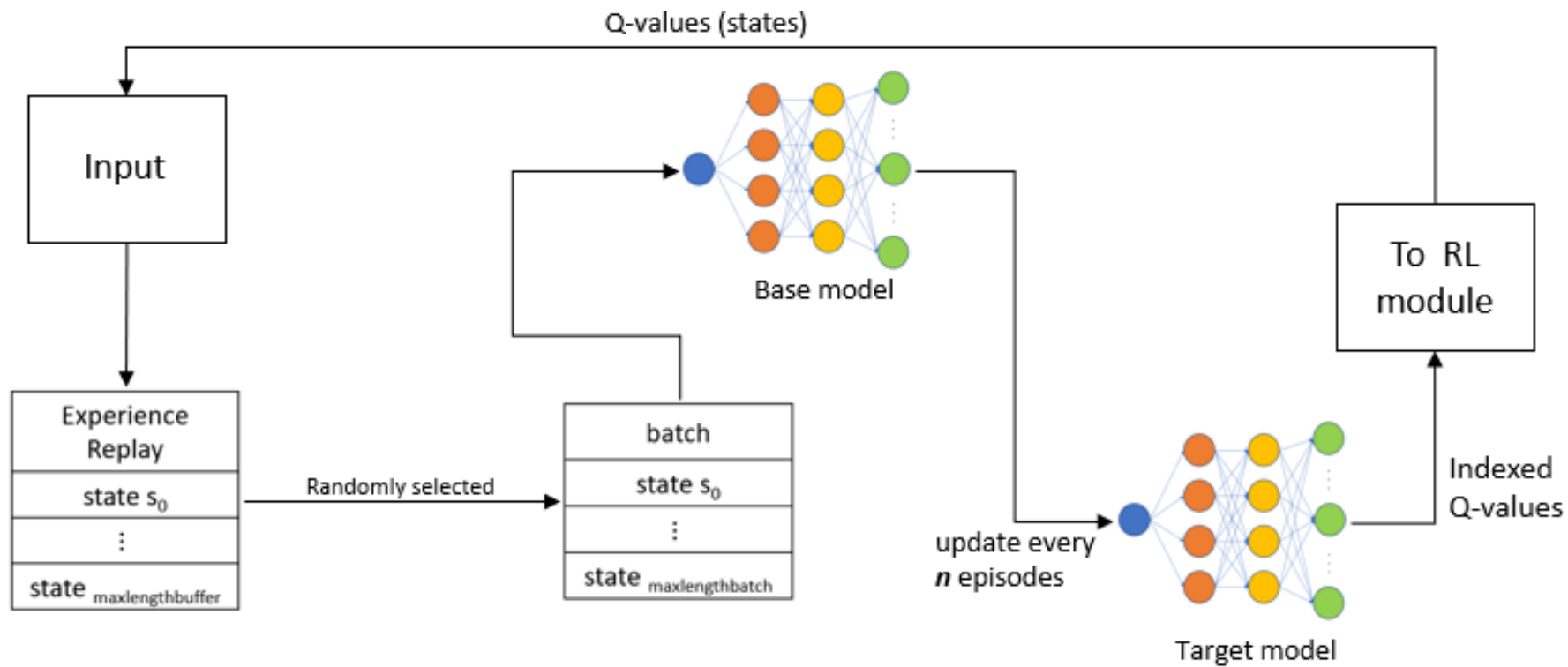


- Some terms:
 - Outputs of the output layer of the NN will be “samples”
 - Buffer: we will randomly save the samples in a buffer
 - The new inputs to the NN will come from the buffer
 - The buffer is now a Replay Buffer
- Replay buffer: Saves “experiences”
 - “Table” that saves random samples to be passed for training
 - Size of the replay buffer varies. It will be a hyperparameter
- Training by batch:
 - Now that we will be using a replay buffer to train, we can make a smaller group of samples. This will be a batch.
 - The NN will use the samples from the batch to train.

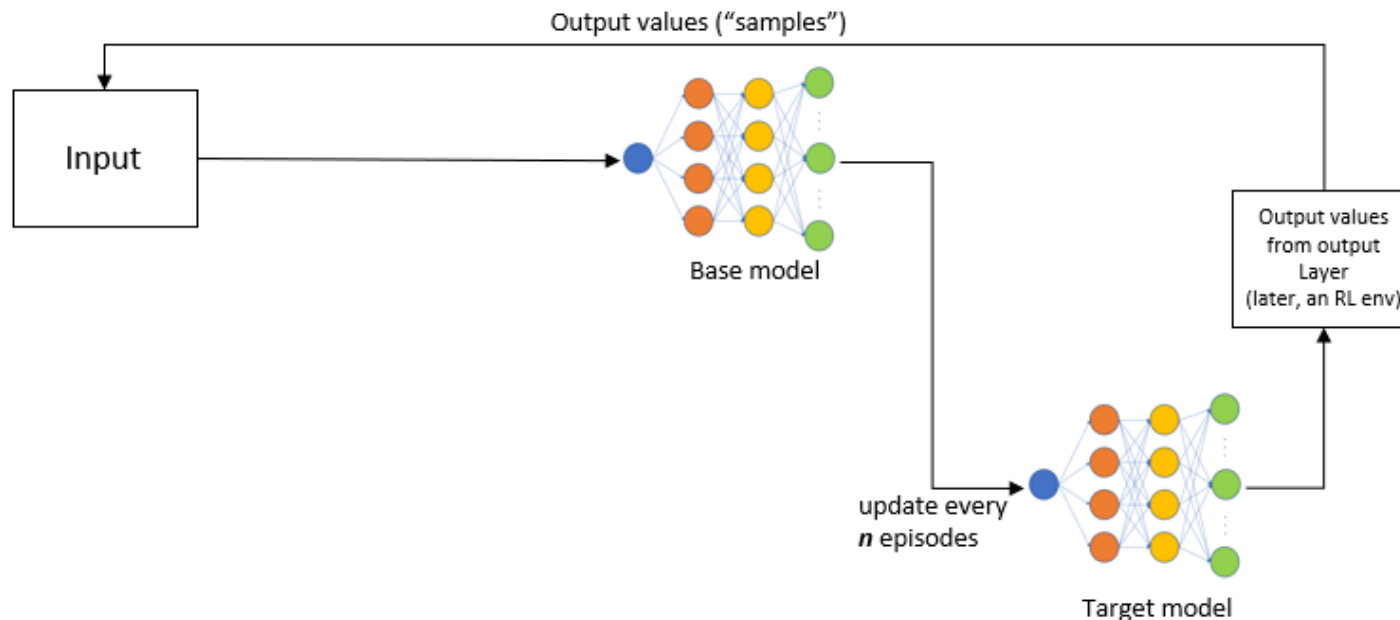




- We will use ER in NN for:
 - **Approximation** of values for:
 - Q-values: indexed.
 - This is Deep Reinforcement Learning



- Training by batch vs single input:
 - The batch contains random experiences
 - The experiences are passed to the NN
 - The NN can continuously be fed from the batch
- Single input
 - One input at a time
 - Weight & bias corrections (optimization updates) are done per input
 - It Takes longer for the NN to update



- Hyperparameters so far:
 - Number of episodes (in NN: “epochs”)
 - Number of steps (In RL, for example: X steps = 1 episode)
 - Epsilon (tied to explore/exploit):
 - Epsilon decay
 - Max & min epsilon
 - Dropout
 - Length of the replay buffer
 - Length of batch (mini-batch)
- Metrics:
 - Accuracy
 - Loss





THE UNIVERSITY OF
SOUTHERN MISSISSIPPI.