Full Hybrid Project Blueprint

Friday, 5 December 2025 9:24 PM

1. Single Neuron Simulation → Output

Steps inside the single neuron model:

1. Start with time steps (dt)
2. Apply input current I(t)
3. Update voltage using:
   - integration (voltage ↑)
   - leakage (voltage ↓)
4. Check threshold
5. If threshold crossed → spike
6. Reset voltage
7. Store:
   - voltage[t]
   - spike[t]

Output:
- Voltage graph
- Spike train (0s and 1s)

2. Build Emotional 2/3 Neuron circuit

Key ideas:
- Neuron A spikes → influences Neuron B
- Influence strength = synaptic weight
- spike arrives with synaptic delay
- Connections can be:
  - Excitatory (increase B's voltage)
  - Inhibitory (decrease B's voltage)

Examples:
- 2neurons: Amygdala (A) → PFC (B)
- 3-neurons: Stimulus → Amygdala (A) → Hippocampus (C) → PFC (B).

This creates emotional patterns

3. Generate Spike Trains for Emotional States

Different emotional inputs = different patterns.

Inputs:
- Calm: low, smooth
- Stress: high + noisy
- Fear: sudden burst
- Excitement: rhythmic/pulsed

Output of simulation:
- spike trains representing each emotion
- These become our dataset

4. Extract Spike features (for ML)

From each spike train, extract:
- Firing rate (how many spikes)
- ISI (Inter-spike Interval)
- ISI variance
- Burstiness
- Rhythmicity
- Entropy (randomness)

These features = numerical fingerprints of each emotion

5. Train ML Classifier

Once features are extracted:

Steps:
1. Create features CSV
2. Use ML models:
   - Random Forest
   - SVM
   - Logistic Regression
3. Train the model on:
   - calm
   - stress
   - fear
   - excitement
4. Evaluate using:
   - accuracy
   - confusion matrix

The model learns how to predict emotion from spike patterns

6. Analyze and Plot Results

Use Matplotlib to plot:
- Voltage traces
- Spike trains
- Emotional patterns
- Feature distributions
- ML confusion matrix
- ML accuracy graphs