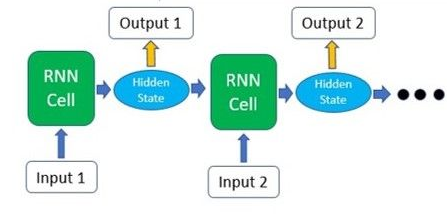
**CSC 417 Unit 2 Day 5 Outline**

1. Recurrent Neural Networks
   1. Introduction
      1. Traditional feed-forward networks accept an input and transform it into an output via a static model based on training data
      2. Each input in a traditional network is analyzed in isolation
         1. Relationships between different inputs are not considered
         2. Past input does not influence the classification of the current input
      3. Traditional networks have difficulty with sequences
         1. Natural language processing
            1. Information is found not only in the words but in their structure (relationships to each other)
   2. RNN Overview
      1. Recurrent neural networks incorporate past input into analysis of the current input
         1. A “hidden state” captures information related to the sequence of inputs



* + 1. RNNs often have a single recurrent layer
       1. Network “depth” relates to the number of inputs processed in sequence
    2. Example: asking a chatbot “what time is it?”
       1. Input first word into network and obtain output
       2. Input second word *and* hidden state into the network and obtain an output
       3. Final output is created based on the inputs *and* the hidden state
          1. Hidden state preserves sequence relationship information
  1. RNN Process
     1. Initialize network layers
        1. Input
        2. Recurrent (often only 1)
        3. Output (feed-forward)
     2. Pass inputs and initial hidden state into network, receive output and modified hidden state
     3. Repeat until all inputs are processed
        1. “Unfolding” a network = processing all inputs in sequence
     4. Predication made via feed-forward layer
     5. Adjustments made via backpropagation through time
        1. Adds series of calculations (linking time steps) to standard backpropagation algorithm
  2. Vanishing Gradients
     1. RNNs have trouble retaining sequence information over many steps (they have “short term” memory)
     2. Later inputs may have a significantly larger impact on the output
        1. A quantity multiplied many times by an amount < 1 can become incredibly small
        2. An RNN contains many time steps – as each input item is processed, derivatives may shrink (vanish)
        3. The more inputs to the RNN, the less impact early inputs will have
           1. Information “reward” is often delayed – early inputs are often important, but this cannot be determined until all inputs are processed
  3. Addressing Vanishing Gradients
     1. LSTM
        1. Stores extra info to “learn” what inputs are important
     2. GTU
        1. LSTM with fewer internal components