*Abstract*— Prediction and perception of temporal sequences for sensory inputs are useful/critical in real-life situations. For sequence learning, a theoretical framework known as HTM (hierarchical temporal memory) is used, which is based on multiple known features of neurons. The HTM model mimics the neocortex's operating principles, which handle sequence learning and storing trained data in memory and can perform prediction operations until the right match is found. The HTM Prediction Engine is evaluated in this paper when applied to a Multi Sequence of Numbers, Sequence of Alphabets (Anti - Cancer Peptide Cells) and Set of Images to predict the sequences of numbers, sequences of alphabets, and images. The primary goal is to examine the HTM prediction Engine and comprehend Multi Sequence Learning for Sequence of Numbers, after which a prediction algorithm for a Sequence of Numbers is developed for predicting a set of sequences that belong to a particular sequence (To Predict Anti-Cancer Peptide Cells) and also to develop Multi Sequence Learning for Numbers can predict whether a specific sequence belongs to the Training Image Sequence Sets.

Keywords—Hierarchical Temporal Memory (HTM), , Homeostatic Plasticity Controller (HPA), Prediction code, HTM Prediction Engine).

# **Introduction**

The ability to perceive and predict temporal sequences of sensory inputs is critical for survival. Hierarchical temporal memory (HTM) sequence memory has recently been proposed as a theoretical framework for sequence learning in the cortex, based on numerous known features of cortical neurons. The model’s sparse temporal codes can robustly handle branching temporal sequences by keeping numerous predictions until enough disambiguating evidence is available.

The medical sciences have advanced to provide us with a significant understanding of the working of the cortex. Investigations have concluded that many cortical regions are part of the temporal sequence processing [1] [2]. On the other hand, ML engineers have been researching sequential memory, leading to several temporal pattern recognition models [3].

Scientists have gained insights by working on the cortex that sequence learning has large invariant changing series of inputs. The exact neural mechanism of sequence memory is still unknown, but models that give a reading of the neurons are used to study. These models show significant capabilities to recollect and recognize the sequence of inputs using rules. These ML models do not match the real-world issues

Hierarchical Temporal Memory (HTM) is a Biomimetics model based on the principles of memory predictions developed by scientists to capture the architectural and algorithmic features of the neocortex [4] [5]. HTM has given promising results in pattern recognition, and This can learn the temporal sequences and spatial flow of sensory inputs as data.

# LITERATURE SURVEY

A. SDRs

Sparse Distributed representations (SDRs) of input patterns serve as the language of HTM. With a set quantity of active bits, it internally creates SDRs. These sentences make sense semantically. The active bits representation in SDR, which is crucial to HTM learning, must be equivalent for two inputs with similar semantic meaning.

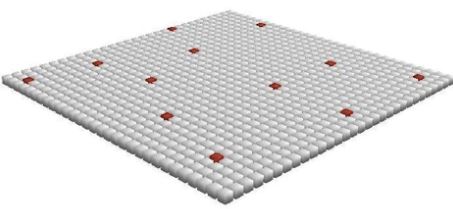


Figure 1 shows the representation of SDR

B. Segments of the proximal dendrite

Little black circles signify synapses, which connect the cells in a column via a proximal dendrite. A genuine synaptic connection with a persistence value over the connection threshold is represented by a solid circle. In contrast, an unfilled circle denotes a potential synaptic connection with a persistence value below the connection threshold. If sufficient valid synapses are linked to active input bits, feedforward input activates a column after a local inhibition step.

## **Hierarchical Temporal Memory (HTM)**

# The HTM model learns what happens in one layer of the cortex. HTM operates on continuous streams of input patterns, attempting to construct infrequent and constant representations of input sequences based on the input stream's repeated pattern.

# HTM's ability to predict future patterns based on previously trained data patterns. HTM receives a unique pattern after a few cycles that compares the previous patterns to the new pattern. Input patterns should not repeat, and the pattern's uniqueness is important for training different sequences of input patterns, providing a wide variety of sequences to be predicted.

# **Methodology**

The project Multi sequence Learning developed using C# .Net Core in Microsoft Visual Studio 2022 IDE (Integrated Development Environment) is used as a reference model to understand the functioning of Multi sequence learning, which uses HTM Prediction Engine and HPA controller for stability.

The objective of this project is to understand Multi sequence learning for the sequence of Numbers and develop Multi sequence learning for Sequence of number Datasets which includes learning of a sequence of data resulting in the prediction of any sequence of numbers based on the learning in HTM.

**Temporal Memory**

**Encoders**(Scalar Encoder/  
HTM Image Encoder)

**Data**

(Sequence of Numbers/  
Sequence of Alphabets/  
Image Datasets)

**HTM Prediction Engine**

**Spatial Pooler**

**Homeostatic Plasticity Controller**

**Prediction Algorithm**

c. Spatial pooler

Spatial Pooler generates SDR input while mapping the cells of the active columns. Each column has a network of synapses connecting it to the next region of input bits. Many columns would look the same, but these columns are distinct. Different patterns generate different levels of activation, and the more robust activation prevents lower activation of the columns. Column areas are adjustable and can range from small regions to the entire area. The inhibitory mechanism is used to provide a skewed representation of the input. A similar pattern results in identical activated columns. HTM learns from its input and unifies cell connections. Learning results from updating synapse permanence. With active bits, the active columns increase the persistence value, while the other columns decrease it. Columns that are not active learn nothing. To ensure that all columns participate in the training, the inactive columns are boosted. The spatial pooler denotes data pools or clusters in the spatial dimension. During the spatial pooler's learning process, each pattern that appears at the input is compared to a database of other patterns.