# Enhancing Semantic Cognition Models with LSTM-Based Question Answering: An Extension of Parallel Distributed Processing

**Rhitvik Sinha**
Courant Institute of Mathematical Sciences, New York University
rs8438@nyu.edu

## Abstract

This report details an extension of the Parallel Distributed Processing (PDP) approach by McClelland and Rogers, aimed at exploring the applicability of Long Short-Term Memory (LSTM) neural networks in semantic cognition. The original model, which simulates stages of cognitive development in recognizing categories such as plants and animals, is expanded with a contemporary LSTM architecture designed for natural language-based question answering. This model is trained to respond to yes/no questions, learning the same facts as its fully connected neural network predecessor. Results demonstrate promising alignment with developmental cognitive stages and suggest the feasibility of integrating connectionist models with natural language processing for enhanced semantic understanding. The code for the experiments is available at `https://github.com/rhitviksinha/ccm`.

## 1 Introduction

Semantic cognition, central to understanding and manipulating linguistic constructs, has been profoundly influenced by connectionist models, notably the Parallel Distributed Processing (PDP) approach detailed by [McClelland and Rogers, 2003]. The approach posits that mental phenomena can be described by interconnected networks of simple units. These networks, often implemented as artificial neural networks [Rosenblatt, 1958], mimic the distributed and parallel way in which the brain processes information. In semantic cognition, this approach suggests that knowledge is represented across different patterns of activation within these networks, and learning occurs through the adjustment of connections between units based on experience. This model has been influential in explaining how people can understand and produce language, recognize patterns, and make inferences based on their knowledge of the world.

Inspired by the capacity of these models to simulate complex cognitive behaviors, this project introduces a novel extension: applying a Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] network to the domain of question answering. This extension aims to test the adaptability of PDP principles within the realm of natural language processing, thereby bridging static categorical differentiation with dynamic, interactive language understanding. Question answering, as a task, presents unique challenges such as the need for understanding context, handling ambiguity, and generating precise answers in real-time, all of which require robust and adaptive neural architectures. LSTMs, a specialized type of Recurrent Neural Network (RNNs) [Bengio et al., 1994], are particularly adept at processing sequences and retaining information over long periods, which makes them exceptionally suitable for such tasks. By leveraging their ability to maintain a 'memory' of previous inputs, LSTMs can effectively handle the sequential and context-dependent nature of language, offering significant advantages in natural language processing tasks such as question answering.

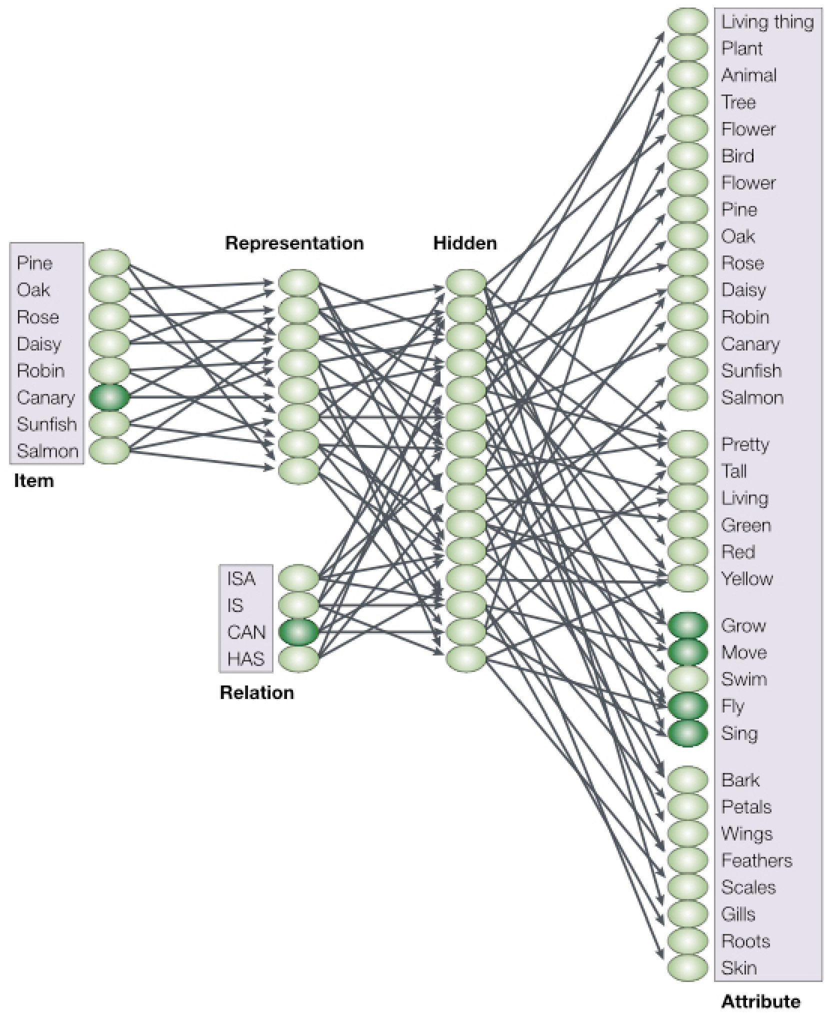


Figure 1: The three-layer artificial neural network for semantic cognition. The network takes "Item" & "Relation" as input, has a "Representation" & a "Hidden" layer, and an output layer of "Attribute"(s). The network models how items like "Pine" and "Canary" relate to specific attributes.

This extension aims to test the adaptability of PDP principles within the realm of natural language processing, thereby bridging static categorical differentiation with dynamic, interactive language understanding. Through integrating the LSTM's capabilities with the PDP framework, this project seeks to enhance our understanding of how deep learning architectures can be modified to accommodate the unique demands of processing and generating language-based responses, providing insights into more adaptive and contextually aware models of cognitive processing.

## 2 Methodology

### 2.1 Original Neural Network Implementation

This neural network model exemplifies a sophisticated approach to semantic cognition by learning and processing semantic relationships and attributes through a distributed network architecture. The model is designed to answer queries that involve an item (e.g., "Canary") and a relation (e.g., "CAN"), subsequently outputting all attributes that accurately describe the item/relation pair (e.g., "grow, move, fly, sing"). The knowledge within this system is stored implicitly in the weights of the connections, starting with random initialization. Through training on comprehensive datasets containing factual semantic relationships, the network employs Stochastic Gradient Descent [Ruder, 2016] to optimize these weights, effectively learning the underlying semantic structures.

#### 2.1.1 Modeling Cognitive Development through Learning

The network's architecture facilitates the modeling of cognitive development in semantic representation through the application of distributed representations and Backpropagation [Rumelhart et al., 1986] learning techniques. During the training process, the model exhibits developmental stages that mirror human cognitive development, showcasing a broad-to-specific differentiation:

- At epoch 500, the model first differentiates between broad categories such as plants versus animals.
- By epoch 1000, it refines these categories into more specific groups, distinguishing between birds versus fish and trees versus flowers.
- The model achieves full differentiation by epoch 2500, representing a detailed and nuanced understanding of various semantic categories.

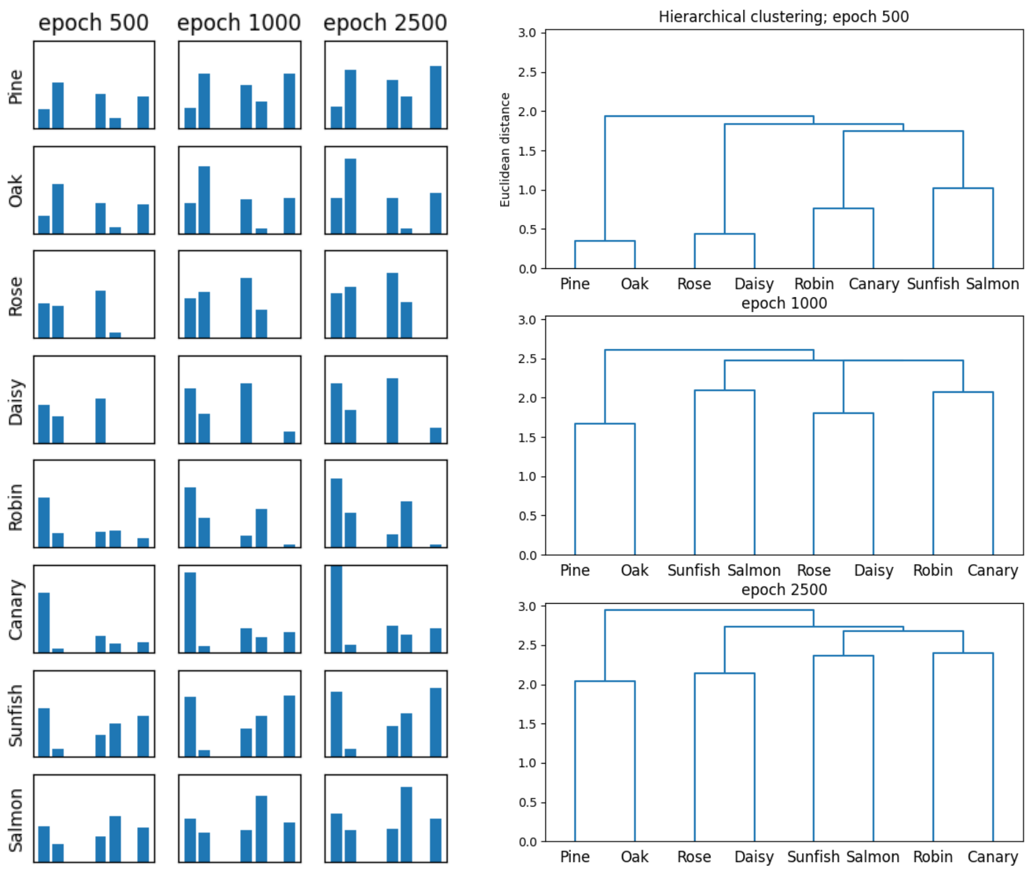


Figure 2: **Pattern of activity over representation layer (left)**: This analysis provides insights into how different semantic concepts are activated across the network during various stages of learning. **Hierarchical clustering of patterns (right)**: By clustering the patterns of activation, the model visualizes the hierarchical organization of semantic knowledge, illustrating how more abstract categories gradually refine into distinct, specific entities.

#### 2.1.2 Key Principles of Neural Network Models of Cognition

The foundation of this model is built on several key principles of neural computation and cognitive modeling:

- Neurally-inspired computation: The design takes inspiration from biological neural processes, where low-level neuronal activities underpin high-level cognitive functions.
- Emergence of intelligence: The model reflects the concept that intelligence is an emergent phenomenon, arising from the complex interplay of numerous simple computational elements.
- Importance of simulation: Given the complexity and the emergent nature of intelligence, simulation through computational models is crucial for unraveling how intricate behaviors and cognitive abilities develop.

This neural network model not only serves as a powerful tool for understanding semantic cognition but also offers profound insights into the patterns of cognitive development observed in humans. By simulating the learning process and visualizing the evolution of semantic understanding, the model underscores the potential of neural networks to decode complex cognitive phenomena, reinforcing the essential roles of neurally-inspired computation, emergent intelligence, and computational simulation in cognitive science.

### 2.2 LSTM for Question Answering

The extension of the neural network model with an LSTM architecture facilitates a more dynamic interaction with natural language inputs, specifically in processing yes/no queries related to the semantic cognition task.

#### 2.2.1 Dataset Generation

The dataset was derived from an original semantic dataset comprising 8 items, 4 relations, and 36 attributes, resulting in 1152 potential data points. Each data point was transformed into a plain text format where item-relation pairs were associated with attributes, leading to the generation of multiple yes/no questions. For instance, for the item “Canary” with the relation “CAN” and attributes such as “sing” and “fly”, the questions generated included “can canary sing” and “Can a canary fly” Initially, the dataset consisted of 1059 data points labeled ‘No’ and 93 labeled ‘Yes’, demonstrating a significant class imbalance. To address this, the minority class (‘Yes’) was oversampled during training to ensure the model learned to predict less frequent outcomes effectively.

#### 2.2.2 Model Description & Training Process

The model uses a 50-dimensional embedding layer to transform textual inputs into dense vectors, followed by an 100-dimensioned LSTM layer to process these vectors sequentially, capturing the temporal dependencies typical of language. The network concludes with a linear layer that maps the LSTM outputs to a binary prediction, processed through a sigmoid function for probability estimation.

The model is trained using the binary cross-entropy loss (BCELoss), optimized with Adam [Kingma and Ba, 2014], an algorithm for first-order gradient-based optimization of stochastic objective functions. The learning rate was set at 0.001. The LSTM model’s architecture, including an embedding dimension of 50 and a hidden dimension of 100, was designed to balance complexity and performance, ensuring efficient training and robust learning capabilities.

## 3 Result

### 3.1 Training Dynamics of the Neural Network

The training of the neural network showcased a developmental trajectory that mirrors cognitive maturation observed in human development. Initially focusing on broad categories such as distinguishing plants from animals, the model progressively honed its capability to delineate finer distinctions, like differentiating between types of birds and types of plants. This staged learning not only highlights the model’s capability to mimic human cognitive development but also underscores the complexity and effectiveness of the underlying neural architecture in capturing intricate semantic categorizations.

### 3.2 Efficacy of the LSTM Model

The LSTM model was tailored to address the challenge of understanding and processing natural language by answering yes/no questions based on learned semantic relationships. After training for 10 epochs, the LSTM model achieved an accuracy of 0.99, precision of 0.99, recall of 1.00, and an F1 score of 0.99. These results indicate a high level of performance in predicting the correct answers, suggesting that the model has effectively learned to associate items with their attributes and relations.

However, it is essential to note the limitations imposed by the dataset's size and the model's complexity. With approximately 1200 samples and a vocabulary size around 50, the model's complexity—characterized by 50-dimensional embeddings and 100 hidden units in the LSTM—might be excessive relative to the amount of training data available. This discrepancy raises concerns about the model's generalizability to new, unseen data despite the high metrics observed during training. The remarkably high performance metrics might also indicate potential overfitting, where the model learns to perform exceptionally well on the training data but may not necessarily replicate this performance on new, unseen datasets.

### 3.3 Potential Limitations

The results from the LSTM model, while impressive, should be interpreted with caution given the potential for overfitting due to the small dataset relative to the complexity of the model. Moving forward, it would be beneficial to explore methods of regularization, such as dropout or early stopping, to mitigate this issue. Additionally, expanding the dataset or simplifying the model could help in achieving a better balance between model complexity and data availability, ultimately leading to a more robust and generalizable system. This exploration into LSTM-based question answering within the framework of semantic cognition not only advances our understanding of neural network applications but also opens avenues for further research into optimizing model architectures for specific cognitive tasks.

## 4 Conclusion & Discussions

This project set out to extend the established Parallel Distributed Processing (PDP) model by McClelland and Rogers through the integration of a Long Short-Term Memory (LSTM) network, aimed at enhancing semantic cognition models with the capability of processing natural language-based question answering. The transition from a conventional neural network to an LSTM model represents a significant stride towards simulating more dynamic and complex cognitive processes akin to human reasoning and linguistic interaction.

The initial neural network demonstrated an ability to mimic cognitive development stages seen in humans, categorizing semantic relationships from broad to specific distinctions through a structured training regime. This provided a solid foundation for implementing the LSTM model, which was specifically designed to handle yes/no questions by learning from a dataset crafted from the original semantic categorization data.

The LSTM model achieved exemplary performance metrics after training for 10 epochs—accuracy, precision, and F1 score at 0.99, and a recall of 1.00. These results indicate that the model successfully learned the semantic connections between items, relations, and attributes within the provided dataset.

While the LSTM's performance metrics are promising, concerns arise regarding the potential overfitting given the relatively small size of the dataset (approx. 1200 samples) and the simplicity of the vocabulary involved (approx. 50 terms). The complexity of the LSTM model, with its 50-dimensional embeddings and 100 hidden units, may not be ideally suited to the scale of data available, potentially limiting the model's generalizability to new, unseen datasets.

## 5 Future Directions

To enhance the robustness and generalizability of the LSTM model, a critical area for future work involves expanding the dataset used for training and evaluation. Expanding the current dataset could involve incorporating more diverse semantic relationships and attributes, or integrating a wider range of items to cover a broader spectrum of semantic cognition. This expansion would provide the model with a richer and more complex array of data points, facilitating deeper learning and more nuanced understanding of semantic categories.

Alternatively, adopting a larger, pre-existing semantic cognition dataset might be beneficial. Utilizing a well-established dataset could not only provide a more extensive base for training but also allow for benchmarking the model against standardized data, offering insights into its performance relative to other models in the field. This approach would help in validating the model's capabilities on a wider scale and in more diverse scenarios, thereby enhancing its applicability and reliability in real-world applications.

## References


Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. doi: 10.1109/72.279181.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. URL `https://arxiv.org/abs/1412.6980`.

James L. McClelland and Timothy T. Rogers. The parallel distributed processing approach to semantic cognition. *Nature Reviews Neuroscience*, 4(4):310–322, 2003. doi: 10.1038/nrn1076.

Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

Sebastian Ruder. An overview of gradient descent optimization algorithms, 2016. URL `https://arxiv.org/abs/1609.04747`.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. doi: 10.1038/323533a0.

## APPENDIX: Data Generation

This appendix provides a detailed description of the methodology used to generate the dataset for training the LSTM model in question answering based on semantic cognition relationships.

The dataset generation begins with encoding items and relations into a binary format. For example, the 12-dimensional input pattern `[ 0 0 0 0 0 1 0 0 0 0 0 1 ]` represents the item "canary" and the relation "has". Each dimension in the input pattern corresponds to a specific item or relation, where a '1' indicates the presence of that item or relation, and '0' indicates its absence.

The 36-dimensional output pattern corresponds to possible attributes associated with the item and relation from the input pattern. This output is used to generate a comprehensive set of question-answer pairs that reflect the possible attributes of the item in relation to the specified relation.

| **Question** | **Answer** |
|---|---|
| has canary living thing | No |
| has canary plant | No |
| has canary animal | No |
| has canary tree | No |
| has canary flower | No |
| has canary bird | No |
| has canary fish | No |
| has canary pine | No |
| has canary oak | No |
| has canary rose | No |
| has canary daisy | No |
| has canary robin | No |
| has canary canary | No |
| has canary sunfish | No |
| has canary salmon | No |
| has canary pretty | No |
| has canary big | No |
| has canary living | No |
| has canary green | No |
| has canary red | No |
| has canary yellow | No |
| has canary grow | No |
| has canary move | No |
| has canary swim | No |
| has canary fly | No |
| has canary sing | No |
| has canary skin | Yes |
| has canary roots | No |
| has canary leaves | No |
| has canary bark | No |
| has canary branch | No |
| has canary petals | No |
| has canary wings | Yes |
| has canary feathers | Yes |
| has canary gills | No |
| has canary scales | No |

Table 1: Sample Question-Answer Pairs Generated from the Semantic Model