University of Waterloo
Faculty of Engineering
Department of System Design Engineering

SYDE 556 – Final Project
Wason Selection Task

200 University Avenue West
Waterloo, Ontario, Canada
N2L 3G1

Prepared by
Sugandha Sharma
ID: 20401693
userid: s72sharm
4B Electrical Engineering
23 April 2015

Table of Contents

List of Figures

List of Tables

1 Introduction

This report describes a cognitive model capable of solving a Wason Selection Task (a four card logic puzzle). The goal of modelling the Wason Task is to make an attempt towards replicating human cognition by producing realistic cognitive behaviour. The model solves the Wason Task in a way which is similar to humans, by using symbolic reasoning and strategy changes across different contexts. This implies that the model learns different behaviours in different contexts and also exhibits structure sensitive generalization (i.e., syntactic generalization).

In general, it is a very challenging problem to create models that produce realistic cognitive behaviour. In addition to that, creating these models with realistically modelled neurons is even more challenging due to the extra computational overhead. However, using realistic neurons allows us to better evaluate cognitive theories. In order to understand how the brain works, it is important that the model should not only produce the correct behaviour as humans, but also do it in the same way as real brains do. This implies that we should be able to see the comparable firing patters and neural connectivity. Moreover, we should also be able to see the same effects of neural degeneration as seen in real brains.

Therefore, for modelling the Wason Task, I have used structure-sensitive computation in an underlying neurally plausible architecture known as the Neural Engineering Framework (NEF) which acts as a neural compiler for building biologically plausible neural models of cognition [1]. This implies that the model can also be used for understanding the effects of neural computation on cognition.

A similar model of Wason Selection Task has been constructed before by Eliasmith, C. (2005) [2]. The model I am creating is an attempt to replicate the functionality of that model and also implement an extension to it by adding an action selection mechanism using the basal ganglia in order to make the model more natural and human like.

2. Background: Wason Selection Task

Wason Selection Task is a four card logic puzzle devised by Peter in 1966. It is one of the most famous task which has been used to study logical reasoning in humans in various semantic contexts.

In Wason Selection task, a conditional rule of the form “if P, then Q” is given to the participants and they are presented with four cards each of which has either P or not-P on one side and either Q or not-Q on the other side. The visible sides of the four cards show P, not-P, Q and not-Q as shown in figure1. The participants are required to identify each of the cards which would have to be flipped over at minimum in order to determine whether the conditional rule provided to them is true for all the four cards. The logically correct answer to this task is to select the cards showing P and not-Q to be flipped over.

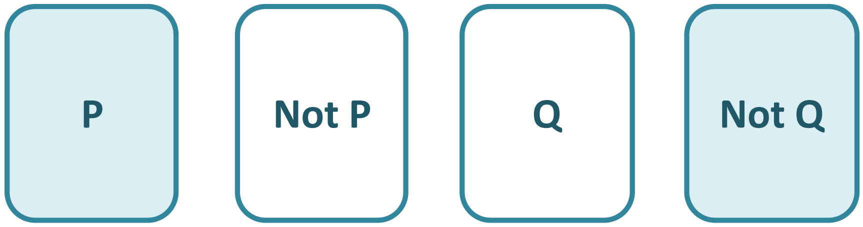

Figure 1. General Form of the Wason Task - If P, then Q

2.1 Content Effect

Research has shown that Wason Selection Task is known to exhibit some interesting content effects which lead the subjects to choose different results based on the semantics (content) of the rule presented to them. Performance in two main general categories of the contexts is explained below:

Abstract Context
When the task is given to the subjects in an abstract form (figure 2), e.g., if a card has a vowel on one side, then it has an even number on the other side, then the majority of the participants choose P (the vowel) and Q (the even number) which is a logically incorrect answer.

Figure 2. Abstract Version - If vowel, then even number

Familiar Context
When the task is given to the subjects in a familiar form (figure 3), e.g., if the person is voting, then the person must be over 18 years old, then the majority of the participants choose P (Vote) and not Q (under 18) which is the logically correct answer.

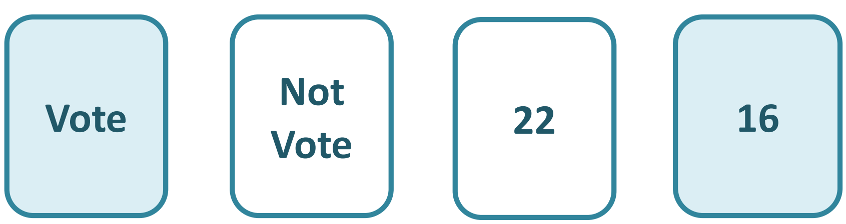

Figure 3. Familiar Version - If voting, then over 18

Thus, structurally identical tasks presented in different contexts lead to different performance. This content effect is probably related to the changes in the symbolic manipulation strategies in the brain based on the content of the task. There are different theories for the explanation of the content effect including familiarity of the contexts, difference in performance in deontic vs non-deontic situations etc. Cheng and Holyoak (1985) suggested that people reason using context sensitive knowledge structures induced from ordinary life experiences. These knowledge structures are called Pragmatic Reasoning Schemas (PRS). Similar to this, the model which I am creating aims to distinguish between the performance on the abstract and familiar contexts using structured representations.

2 System Design

2.1 System Description

2.1.1 Neuroanatomical Mapping

To begin with, it is essential to suggest which anatomical structures might be performing the relevant functions in the model. Figure4 shows how the model is mapped to the functional anatomy.

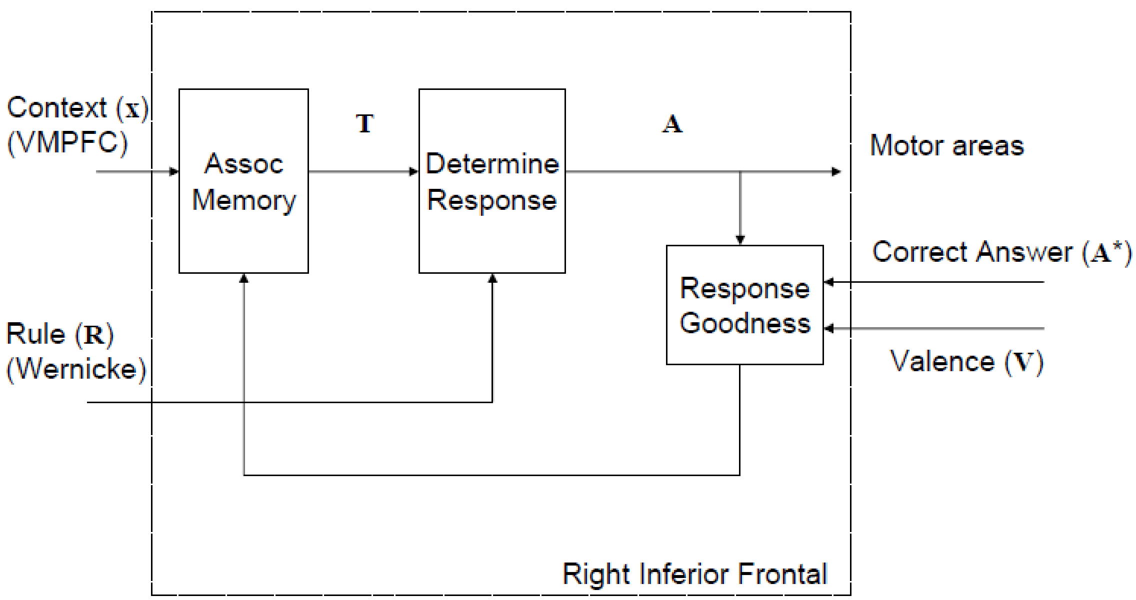

Figure 4. Functional decomposition and anatomical mapping of the model (The letters indicate the vector signals in the model associated with the area shown) [2]

- Ventromedial Prefrontal Cortex (VMPFC) – Provides familiarity or context information that is used to select the appropriate transformation (based on familiar or abstract contexts).

- Wernicke – These are the left language areas which provide the representations to the rule which needs to be examined by the model.
- Anterior Cingulate Cortex (ACC) – It provides an error signal which consists of either the correct answer or some indication regarding whether the response was correct or not. This is then used to learn the correct response give a specific context.
- Right Inferior frontal Cortex – This is what is modelled by the core Wason Network. In this area, the contextual information from VMPC is combined with the linguistic information from the Wernicke to select and apply the appropriate transformation in order to solve the Wason Task. During the application of this transformation, associative learning occurs in the associative memory. This learning occurs based on the response goodness which is dependent on the difference between the response obtained and the correct response indicated by ACC.

2.1.2 Representations

The model has been created in a biologically plausible way by using the Neural Engineering Framework (NEF) [1]. Different anatomical components discussed above are modelled by populations of neurons which represent their respective properties. The mathematical description of encoding and decoding for a vector x in a population a consisting of neurons a_i is given by the equations below. The vector x is used to model the stimulus and the neuron population uses the spiking activity in order to represent this stimulus. In the Wason Model, the stimulus is always a 32 dimensional vector.

Encoding Equation:

$$a_i(t) = \sum_{n=1}^{N} \delta(t - t_{in}) = G_i[\alpha_i \langle x.\Phi_{im} \rangle + J_i^{bias}$$

δ_i (.) are the N_t spikes at time at time t_n for neuron a_i. These spikes are generated by the spiking non linearity G_i in the population with N neurons. G_i Indicates the neuron model being used which in this case is chosen to be the Leaky Integrate and Fire Neuron.

The neuron parametersα_i, Φ_i and J_i^{bias} are the gain, preferred direction vector or a neuron in the stimulus space and the bias current of the neuron respectively.

Decoding Equation:

$$xhat \quad = \sum_{i=1,n=1}^{Nt,N} h_i(t - t_{in})\Phi_i^{\,x}$$

The $h_i(t)$ are the linear decoding filters which are chosen to be the Post Synaptic Currents generated in the subsequent neuron's dendrites. This ensures biological plausibility. $\Phi_i^{\,x}$ (the preferred direction vector) determines the importance of a particular neuron's response to the estimate of x.

Holographic Reduced Representations (HRR)

Holographic Reduced Representations (HRR) have been used to integrate the structure sensitive processing in the model. The model encodes structure in the form of distributed vector representations using circular convolution for vector binding. Moreover, in order to decode the structure, circular co-relation is used. This is done using the semantic pointer architecture (SPA)[3]. The operations are defined as follows:

$$C = A \otimes B \quad \text{and} \quad B \sim A \otimes C'$$

In the above equation, co-relation has been defined in terms of convolution where ' indicates an approximate inverse. This is how it has been implemented in the model. The mathematical description of these operations are as follows:

$$c_j = \sum_{k=0}^{n-1} a_k b_{j-k} \quad \text{And} \quad b_j = \sum_{k=0}^{n-1} a_k c_{j+k}$$

This convolution operation is built into the Nengo2 open source software using NEF, which was used to implement this model.

An example of how concepts and rules are represented in the model is given below:

Rule: *If you can vote, you must be over 18 years of age.*

Here, vote is a concept which is given some semantics to define it, implying that voting is an action having a particular age as its prerequisite.
Vote = ACT*VOTE+PREREQUISITE*AGE

The rule can thus be defines as follows:
RULE = ANTE*ACT*VOTE+PREREQUISITE*AGE + REL*IMPL + CONS*OVER18

Where: Antecedent = Vote Consequent = Over 18
The relationship between the antecedent and the consequent is that antecedent implies the consequent (if antecedent is true, the consequent must be true).

The correct answer assuming that this is a familiar context would be represented as:
ANSWER = VOTE + NOT*OVER18

2.1.3 Learning Network

The learning network used in the model is shown in figure 5. The circles in the figure indicate various populations of neurons and their labels indicate what each of them represent.

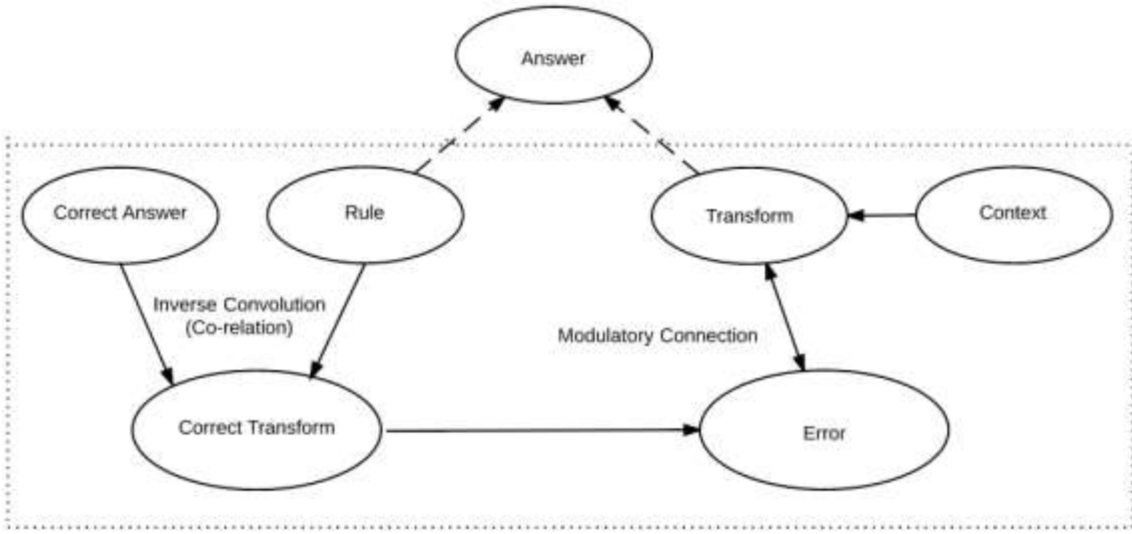

Figure 5. Network Structure for Associative Learning

The transformation in a given context is learnt by changing the connection weights between the neurons in populations representing the 'Context' and 'Transform'. The circular co-relation between the correct answer and the rule are used to compute the correct transform. Learning then takes place based on the error between the 'Transform' and the 'Correct Transform'.

The learning rule which is being used in the model is the Homeostatic Prescribed Error Sensitivity (hPES) rule [6]. This rule is a combination of PES (Prescribed Error Sensitivity) and BCM (Bienenstock, Cooper, Munro learning) rules.

The PES rule was proposed by MacNeil and Eliasmith (2011). This rule minimizes the error between the response obtained and the required response online.

$$\Delta\, d_i = \kappa E a_i$$

where κ is a scalar learning rate, and E is a vector representing the error we wish to minimize. This rule minimizes the error and adjusts the connection weights accordingly by translating a global error signal to a local error signal that can be used to change an individual synaptic connection weight.

When put in terms of connections weights (w_{ij}), the rule resembles backpropagation.

$$\Delta\, w_{ij} = \kappa \alpha_j e_j . E a_i$$

The quantity $\alpha_j e_j \cdot E$ is analogous to the local error term δ in backpropagation are both a means of translating a global error signal to a local error signal that can be used to change an individual synaptic connection weight.

However, the key difference between this rule and backpropagation is that the global-to local mapping is done by imposing the portion of the error vector space each neuron is sensitive to via its encoder. This limits flexibility, but removes the dependency on global information, making the rule biologically plausible.

The general form of BCM rule is as follows:

$$\Delta w_{ij} = \alpha_i \alpha_j (\alpha_j - \theta)$$

Where θ is the modification threshold. BCM rule is based on the intuition that cells driven above their expectation must be playing an important role in a circuit, so their afferent synapses become potentiated. Cells driven less than normal have synapses depressed. If either of these effects persists long enough, the modification threshold changes to reflect the new expectation of the cell's activity. This rule helps to accomplish weight sparcification.

A deficiency of PES rule is that it lacks biological plausibility since biological synapses can change when no error signal is present. More practically, transformation learning may be easier in more sparse systems. For these reasons, hPES (i.e., PES combined with BCM) rule leads to a combination of the error-minimization abilities of the PES rule with the biological plausibility and sparsification of the spiking BCM rule. This rule is already implemented in a biologically plausible way in nengo2.

2.1.3 System Architecture

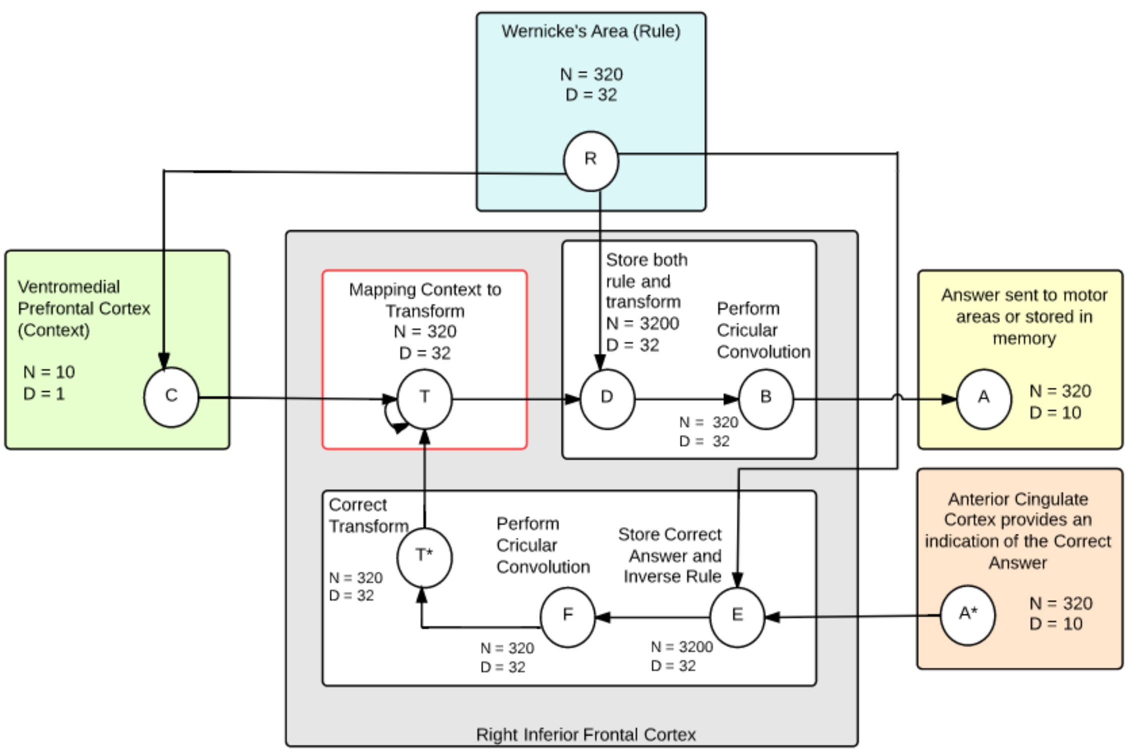

Figure 6. The Core Network Architecture of the Wason Model

The network architecture and core function of the Wason Model is shown in figure 6. In the figure, the circles indicate neural populations. N is the number of neurons and D is dimensionality of the value represented by the neurons in a particular population. Arrows indicate neural connections. A description of the various populations in the model is given below:

R – This population represents the input Rule provided to the system.
C - This population represents the context determined based on the semantics of the input rule.
T – Transform (reasoning strategy) computed by the model to determine the correct answer. This population is recurrently connected to provide an error signal for the learning rule. This is the population (indicated by a red rectangle) where the associative learning occurs in the system.
D, E – This is where the reasoning strategy is applied by computing the convolution of the rule and the transform.
A – This is the result of the transformation or the final answer output given by the model.
A* - This is the feedback which indicated the correct answer.
E, F – This is where the relationship between the correct answer and the rule is determined to compute the correct transform.
T* - This population represents the correct transform determined by the model.

System Functional Relations

Table 1. System functional relations for the Wason Model

Function	Explanation
$T^* = A^* \otimes R'$	The correct transform is computed by using the correct answer and the rule to determining the circular co-relation between them.
$E = T^* - T$	Error signal used to modulate the connection weights in order to map the context to the transform.
$A = T \otimes R$	Correct answer is computed by the circular convolution of the Transform and the input rule.

Action Selection by the Basal Ganglia

The core network of the Wason Model was further extended to include an action selection mechanism. This was done in order to make the model more natural and similar to humans. Action selection (choosing a particular action out of the various possible actions available) is a fundamental cognitive process, and is widely believed to involve the basal ganglia.

The model gets a sensory input which determines the current state of the cortex. The basal ganglia maps the different states to the courses of action and projects it on to the thalamus which then projects the selected action (selected based on its utility) to the right inferior frontal cortex (shown in figure 6) where it gets processed.

The mechanism makes it possible for the model to be more realistic and natural since it is now possible to show the model an abstract rule and a familiar rule and then ask it for the answer to these at a later point in time. Thus the model is able to determine the answers which it thinks are right and store them in it's memory unless specifically asked for the answers. Hence the output from the neural population A (figure 6) is stored in the memory and is provided as an output only when the experimenter asks for the answer at a later point in time.

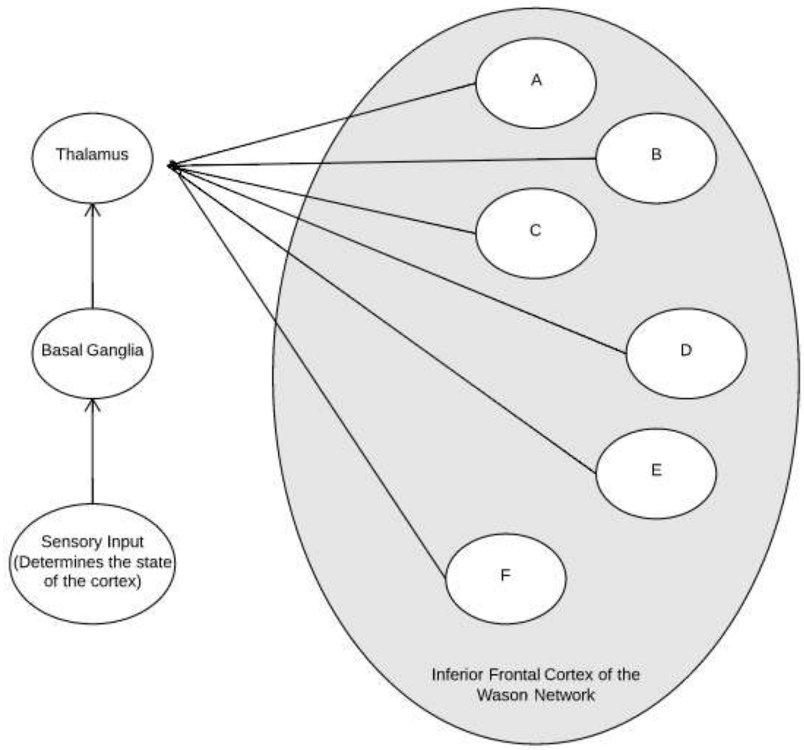

Figure 7. Action Selection by the Basal Ganglia

The different states are shown in figure 7 are explained below:

A – This is the learning state where the model is told to learn from the examples presented to it. This is the only state during which the learning is turned ON.
B – This is the resting state where the leaning is turned off. The model will be in this state when no input rule is presented to it.
C – This is the state when the model is presented with an abstract rule. Whether the rule is abstract or familiar is determined by the cortex. The model computes the answer using the context and stores it in memory.
D - This is the state when the model is presented with a familiar rule. The model computes the answer using the context determined by the cortex and stores it in memory.
E – This is the state when the model is asked for an answer for the previous abstract example which was presented to it. The model outputs the answer from the memory.

F - This is the state when the model is asked for an answer for the previous familiar example which was presented to it. The model outputs the answer from the memory.

2.2 Design Specifications

In order to make sure that this model made of LIF neurons is similar to natural neural systems, it is important to be familiar with the characteristics of the real neurons. Studies have shown that the real neurons have 10% error while transmitting information (Neurons are imprecise). Thus I assume a possibility of a potential 10% error in information transmission through the neural connections. The neuron parameters used for the model are mentioned below:

Maximum Firing Rates: Between 200 to 400 Hz

The firing rate of neurons in different areas of brain are different and they also depend on how much activity that part of the brain is currently undergoing [5]. The maximum firing rates for the purpose of the model have been chosen to be between 200 to 400 Hz since I have assumed that the areas being modelled are in a functional state for the most of the time when the model is run and therefore fire actively.

RC time constant (tau_rc) = 20ms

The membrane RC time constant is the product of the membrane's resistance and capacitance. As tau_rc increases, the slope of the LIF curve increases which means that there is greater change in the firing rate of the neurons for a given change in the stimulus. In general tau_rc lies around 10 – 20ms and for the purpose of this model, a value of 20ms was chosen.

Absolute Refractory Period (tau_ref) = 2ms

tau_ref is the refractory period which is the amount of time it takes for an excitable membrane to be ready for a second stimulus once it returns to its resting state following excitation (recovery time). Thus basically, tau_ref represents the minimum time between consecutive spikes. It is chosen to be 2ms since based on research conducted tau_ref usually lies between 1-2 ms[4].

Filters

Post Synaptic Currents (PSCs) are being used for filtering since they compare very favourable to the optimal filters computed mathematically [1]. Decoding with PSCs instead of using optimal filters results in a small reduction in information transmission, but a large gain in neural plausibility. However, increasing the number of neurons can make up for the reduction in the coding accuracy.

$$h_{psc}(t) = e^{-t/\Gamma_{syn}}$$

Where Γ_{syn} is the post synaptic time constant chosen to be 5ms for all the neural connections except at the output connections which are used to examine the model output. Here the PS time constant is chosen to be 100ms to get an averaged output which can be clearly examined.

Number of neurons and dimensions
Number of dimensions = 32
Number of Neurons for Representation = 320
Number of Neurons for Convolution = 3200
Number of Neurons for Memory = 1280

Higher number of neurons were chosen for populations computing the convolutional functions since it is expected that a higher number of neurons would be required in the brain to compute such a function. Representational populations contain the least number of neurons since there is not computation involved and the memory populations have higher number of neurons than representational population since they are essentially integrators being used for remembering the values.

Memory
It is assumed that the contents of the memory stay stable (no decay) for a short duration of the simulation (11 seconds). This is a reasonable assumptions since though humans have a decaying memory and forget information over time, this experiment is conducted over a short interval of time which does not seem to be enough to encounter significant memory decaying effects. This is because if I give a particular Wason Problem to a person, I will likely ask them what they thought the answer was within the next few minutes (and not after a few days). Since I am using memory to store the answer only until the experimenter asks the subject (i.e., the model), so this assumption is reasonable.

2.3 Implementation

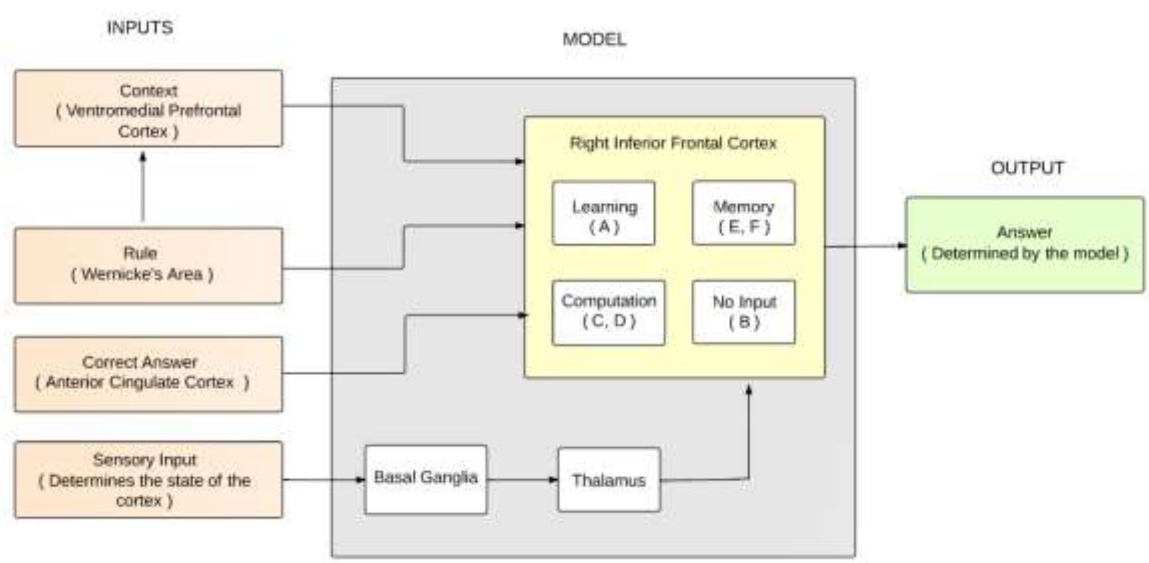

Figure 8. Block Diagram showing the inputs, output and the key components of the Model

Figure 8 shows a high level block diagram of the model showing the inputs and the outputs. The diagram gives an overall view of the integrated model for better clarity about the structure of the entire system. I have limited the scope of the Wason Model to model only the components included in the grey box. In addition to that, although, the context is an input to the core model, it has also been explored to some extent how the context determination might be taking place in the Ventromedial Prefrontal Cortex based on the incoming rule from Wernike's Area.

Note that after several experiments, specific parameters were chosen for the model to confirm that the model behaved as expected (listed under design specifications in Section 2.2). Once these parameters were chosen, the model was run in direct mode (without using neurons) except for the learning population which was run in LIF mode (using leaky integrate and fire neurons), for some of the experiments which were just used to show that the model worked computationally. Some other important experiments were done in LIF mode specially for evaluating the model's performance under various perturbations. This was done to save computation time. Moreover, the performance of the model when run entirely in LIF mode is found to be almost same as when run in the direct mode since the neuron parameters used fit with the model in such a way that the neuron saturation does not limit model's performance.

The model has two experimental modes (Learning and Syntactic Generalization), which make it easier to examine and analyse it's behaviour. These two modes are described in the next sections.

2.3.1 Learning

Goal: To demonstrate that the model can learn the reasoning strategy corresponding to the abstract and familiar versions of the Wason Task just as humans do. Moreover the model also behaves like humans by providing the answer to a question when asked for it at a later point in time relative to when the question was asked.

In this mode, the model is shown an abstract and a familiar rule for 1.5 seconds each. The model learns the transformation corresponding to these rules over the first 3 seconds. Later the model is shown the same rules again and asked for the answers. The model determines the reasoning strategy based on the context of the rule presented to it and ascertains the answer accordingly. The context of the presented rule is determined using a similarity measure (dot product) between the rule being presented to the model (test rule) and the two rules which it has already seen before. If the test rule is similar to the familiar rule which the model was trained on, it is considered as a familiar context, otherwise, it is considered as an abstract context.

The transition of the model through various stages is explained in table 2.

Table 2. State transitions during Experimental Mode1

Simulation time	Cortex State	Behaviour of the Model
t <= 3	A	Given the two rules, their contexts and the correct answers, the model should learn the transformation for mapping the context to the transform.
3 < t <= 4.5	C	State where an abstract rule is given to the model. The model determines the answer and stores it in its memory
4.5 < t <= 6	D	State where a familiar rule is given to the model. The model determines the answer and stores it in its memory
6 < t <= 7	B	No Input, resting stage with no learning
7 < t <= 8.5	E	State where the model is asked for the answer for the abstract rule that it had seen before. The model will output the computed answer stored in its memory.
8.5 < t <= 9	B	No Input, resting stage with no learning
9 < t <= 10.5	F	State where the model is asked for the answer for the familiar rule that it had seen before. The model will output the computed answer stored in its memory.
t > 10.5	B	No Input, resting stage with no learning

The transitions that the model goes through are also illustrated pictorially in figure 9 which shows the utility of different actions in the cortex and different points in time, based on which the actions are selected by the basal ganglia.

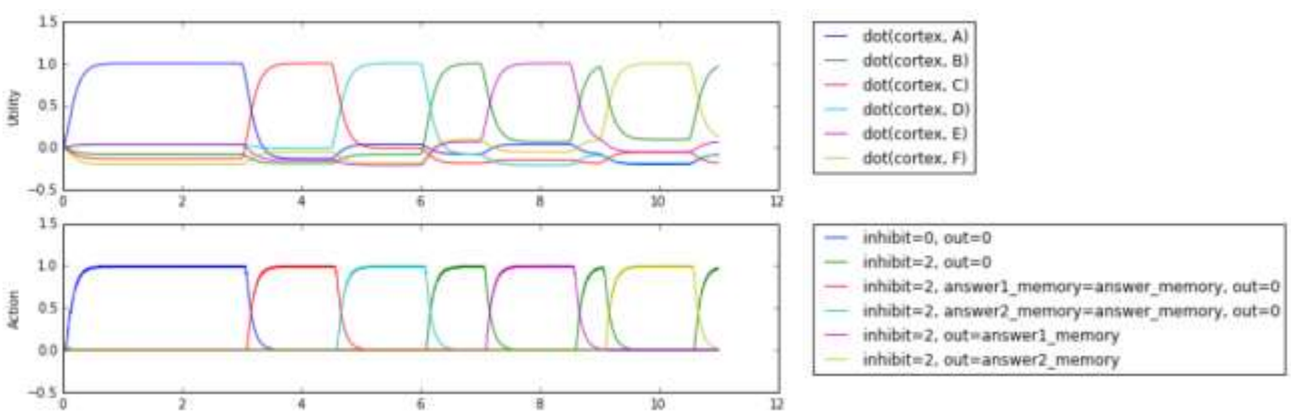

Figure 9. Utility and Action selected by the basal ganglia in Experimental Mode 1

Results for Experimental Mode 1

In figure 10, the model was presented with an abstract rule for the first 1.5 seconds, and then a familiar rule for the next 1.5 seconds. Following are the abstract and familiar rules presented to the model in this experiment:

Abstract Rule: *If a card has a vowel on one side, it must have an even number on the other side.*
Correct Answer: VOWEL + EVEN
Familiar Rule: *If you can vote, you must be over 18 years of age.*
Correct Answer: VOTE + NOT_OVER18

The model learns the transformations corresponding to these rules and does a good job of computing the correct answer when presented with the same rules again (abstract rule when 3 < t <= 4.5 and familiar rule 4.5 < t <= 6). The computed answers are stored in the memory (shown in figure11) and after t = 6, there is no more computation.

Recall that it is assumed that the contents of the memory stay stable (no decay) for a short duration of the simulation (11 seconds) as discussed under Design Specifications (Section 2.2).

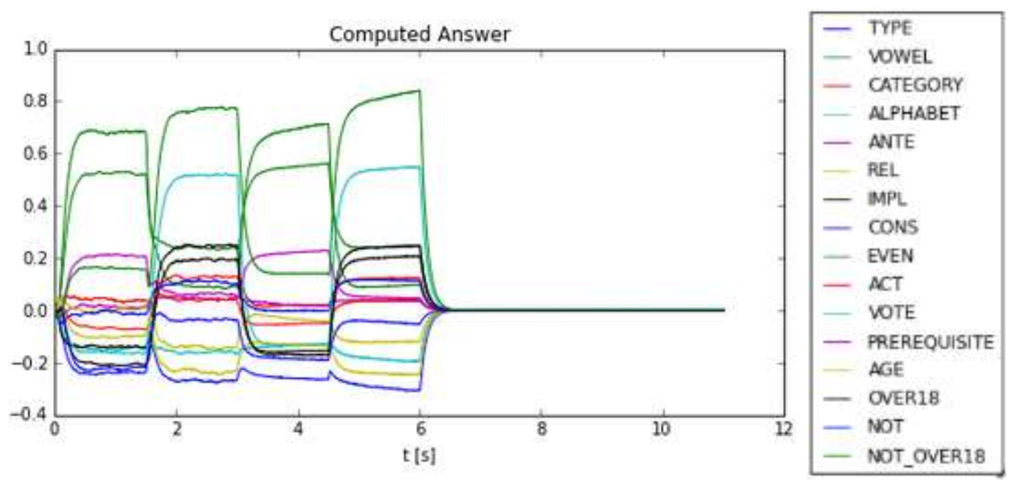

Figure 10. Computed answer which is stored in memory

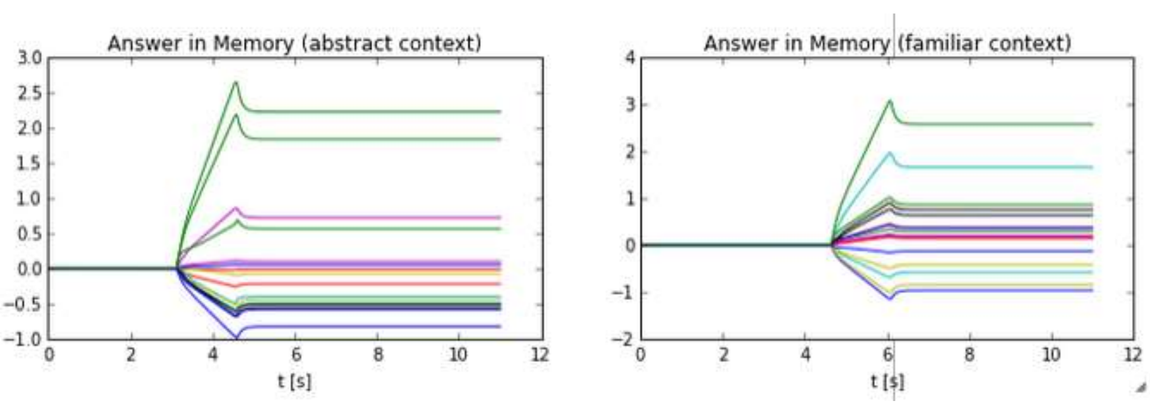

Figure 11. Answers stored in the model's memory

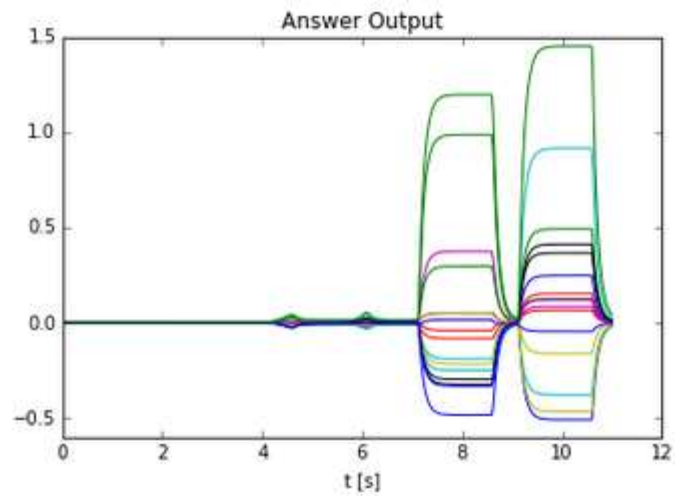

Figure 12. Answer output given by the model when specifically asked for a response

Figure 12 shows the answer output given by the model. Note that the model doesn't output anything unless the experimenter asks for an answer for a particular rule that was presented to it before. Here, the model was asked for the answer to the abstract rule when $7 < t <= 8.5$ and for the familiar rule when $9 < t <= 10.5$. Note that the answers to both these contexts had already been computed before t = 6 seconds, however the model responds only when asked to do so, making its behaviour similar to that of humans.

2.3.2 Syntactic Generalization

Goal: To demonstrate that the model is able to generalize its learnings/reasoning strategy to other similar rules presented to it. Moreover, it should not only determine the context of the new rule but this inference should also improve with the increase in the number of example rules presented to it during the learning phase.

Context Similarity

The context is determined using a similarity measure (dot product) between the rule being presented to the model (test rule) and the rules which it has already seen before. An average of the similarity measures with different example rules presented to the model is computed and scaled by a factor which accounts for the number of similar rules which the model has seen before. This allows to account for the fact that the performance should improve with an increase in the number of rules belonging to the same context which the model has seen before.

This experimental mode is further segregated into three sub-modes: Sub-mode2, Sub-mode3 and Sub-mode4 as explained below:

Sub-mode2 (default-mode): Two rules belonging to the same context are presented to the model for 1.5 seconds each in order to allow the model to learn the reasoning strategy for this particular context. Then a third rule (test rule) different from the first two rules but having the same context is presented to the model. The model determines the context and generalizes its learning to answer the Wason Problem based on the test rule presented to it.

Sub-mode3 and Sub-mode4: These modes are similar to the Default-mode except that three and four example rules respectively are presented to the model during the learning phase. It is expected, that the performance of the model in determining the context and applying the reasoning strategy should improve as the number of example rules which it has seen (during the learning phase) increase.

The state transitions for this mode are explained in table 3. Note that during the leaning phase (i.e., state A), each example rule is presented to the model for 1.5 seconds.

These transitions are also illustrated pictorially in figure 13 which shows the utility of different actions in the cortex at different points in time, based on which the actions are selected by the basal ganglia during this experiment.

Table 3. State Transitions during Experimental Mode2

Simulation time	Cortex State	Behaviour of the Model
t <= 6	A	Given multiple rules (two, three, or four), their contexts and the correct answers, the model should learn the transformation for mapping the context to the transform.
6 < t <= 7.5	D	State where a familiar rule having the same context as the previous rules is given to the model. The model determines the answer and stores it in its memory
7.5 < t <= 8.5	B	No Input, resting stage with no learning
8.5 < t <= 10	F	State where the model is asked for the answer to the familiar rule presented to it. The model will output the computed answer stored in its memory.
t >= 10	B	No Input, resting stage with no learning

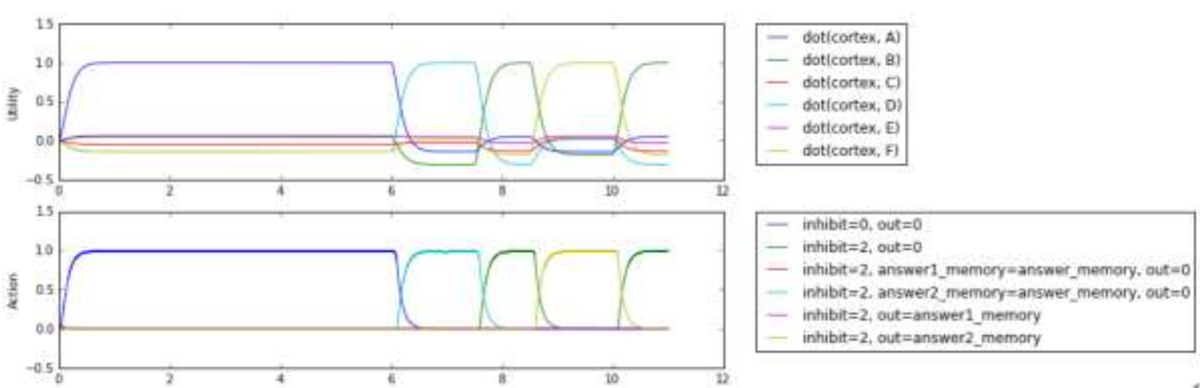

Figure 13. Utility and Action selected by the basal ganglia in Experimental Mode 2

Results for Sub-mode2 (Default-mode)

Rule1 and rule2 are presented to the model one at a time for the first 3 seconds. The model learns the transformation which it then applies to a third rule presented to it at 6 < t <= 7.5. The model computes the answer and stores it in the memory. It then outputs the answer when asked for it by the experimenter at 8.5 < t <= 10

Rule1: *If you drink, you must be over 21 years of age.*
Correct Answer: DRINK + NOT_OVER21
Rule2: *If you can vote, you must be over 18 years of age.*
Correct Answer: VOTE + NOT_OVER18
Rule3 (Test Rule): *If you drive, you must be over 16 years of age.*
Correct Answer: DRIVE + NOT_OVER16

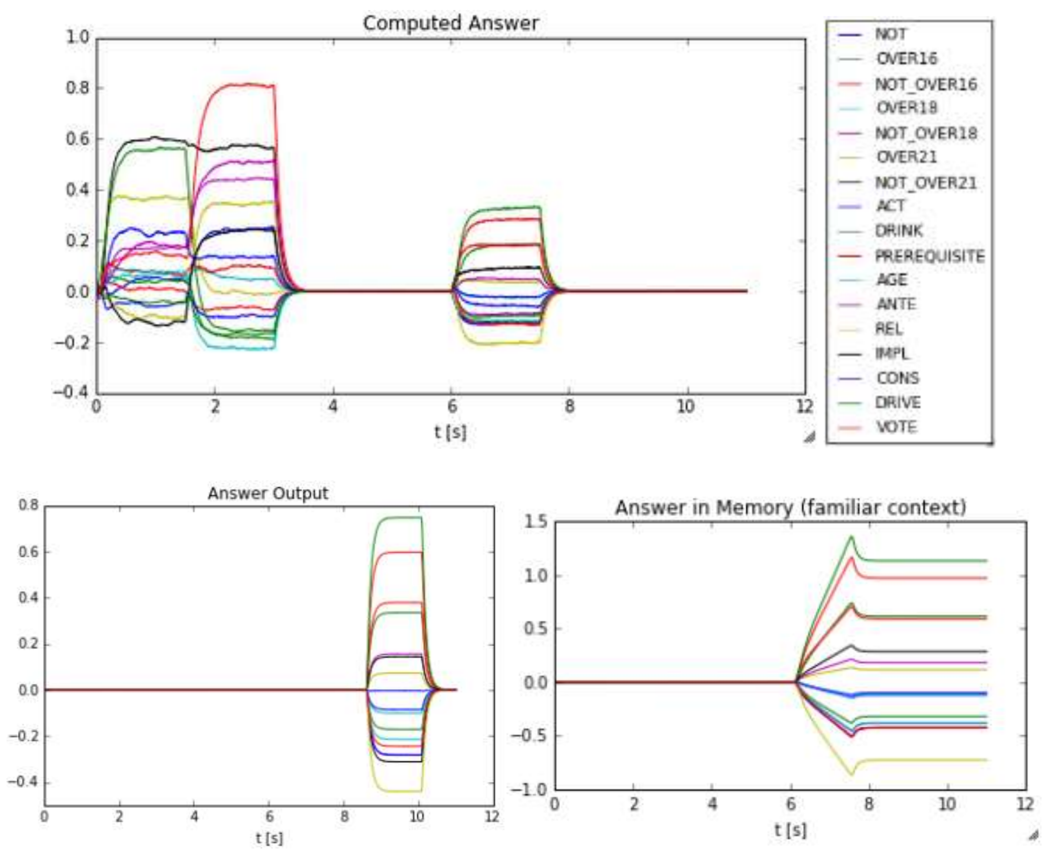

Figure 14. Results for the Sub-mode2 in Experimental Mode2

Results for Sub-mode 3

Same steps as default mode are followed except that one additional rule (Rule4) is presented to the model during the training period (i.e., three rules for learning).

Rule4: *If you are going for skydiving, you must be over 16 years of age.*
Correct Answer: SKYDIVING + NOT_OVER16

Same test rule is used as in the default mode. Figure 15 shows the results of this experiment.

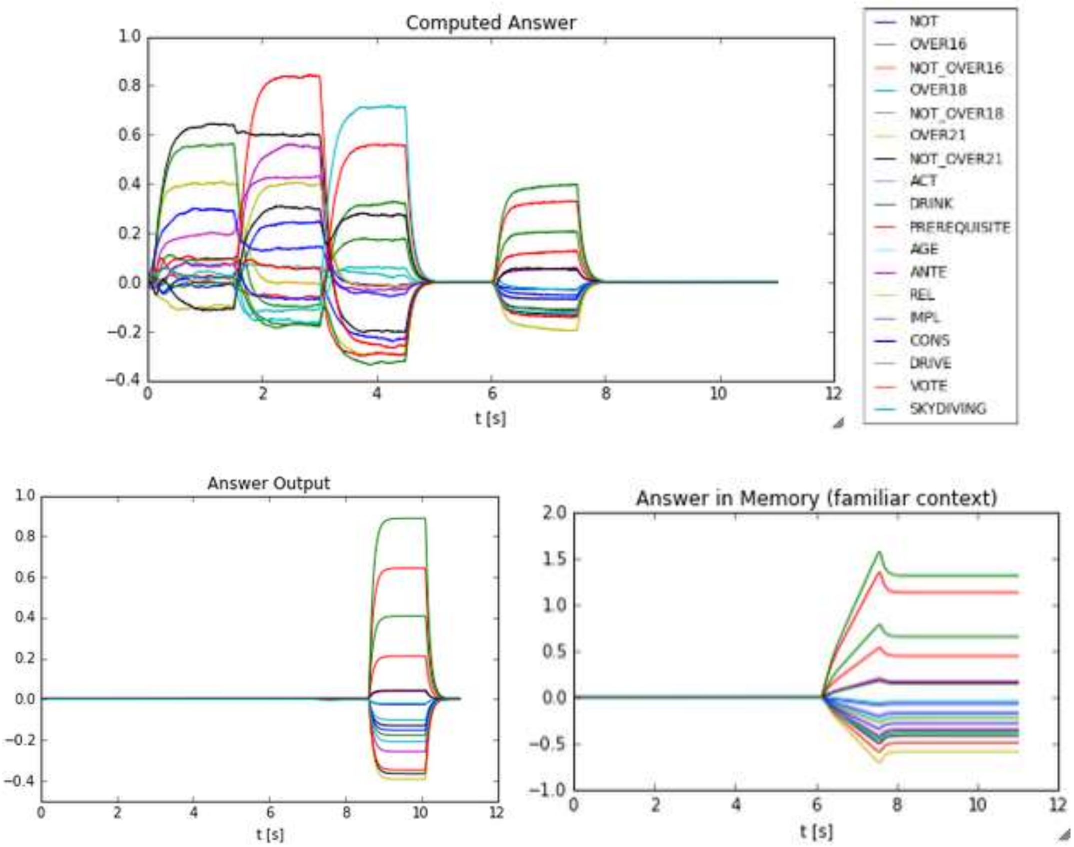

Figure 15. Results for the Sub-mode 3 in Experimental Mode2

Results for Sub-mode 4

Same steps as Sub-mode3 are followed except that one more additional rule (Rule5) is presented to the model during the training period (i.e., four rules for learning):

Rule5: *If you are applying for Canadian Citizenship, you must be over 18 years of age.*
Correct Answer: CITIZENSHIP + NOT_OVER18

Same test rule is used as in the default mode. Figure 15 shows the results of this experiment.

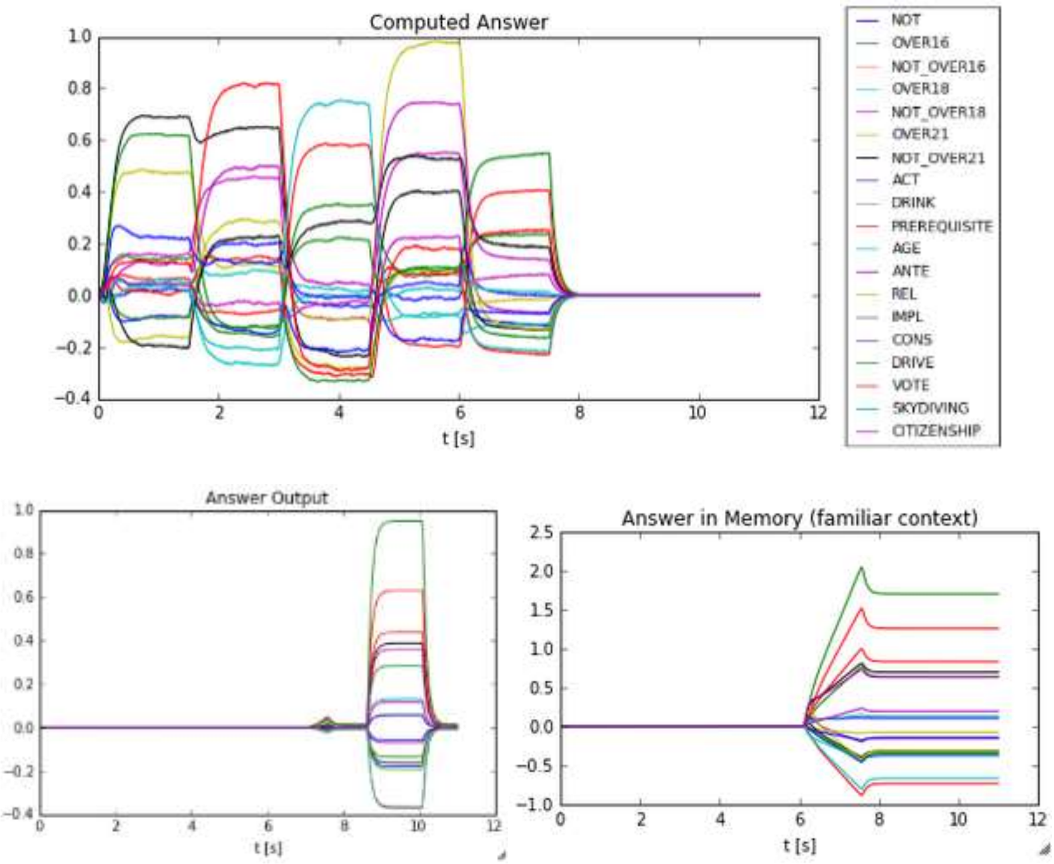

Figure 16. Results for the Sub-mode 4 in Experimental Mode2

Discussion

The model was successfully able to generalize and provide the correct answers in the most of the scenarios in all the three Sub-modes, however the accuracy over a number of trials was found to be different for different modes (table 4).

From the plots of the "Computed Answer", it can be seen that the answer for the test rule computed at 6 < t <= 7.5, is more clearly distinguishable in the Sub-mode4, relatively little less distinguishable in Sub-mode3 and the least distinguishable in the default mode (Sub-mode2). Moreover, over a number of trials it was observed that the accuracy was higher for the Sub-mode4 and Sub-mode3 for which no instance of failure to generalize for the test rule was observed, whereas there was one instance of failure observed for Sub-mode2 over 5 simulations. It can also be seen that in the Sub-mode2, figure 14 there was a slight error in learning the answer to the second rule presented to the model. However the error was so minimal that the model was able to still generalise correctly, though the answer was relatively less distinguishable. Similar error also occurs in Sub-mode3 (figure 15), and Sub-mode4 (figure 16), but its effect on the answer is reduced due to more number of rules presented to the model for learning. This is confirmed by the generalization accuracy shown in table4. Thus errors while learning phase are more likely to effect the generalization if only two rules are presented and the likelihood of error in the generalization decreases as the number of rules presented for learning increase.

Note that this particular error caused in learning the answer for the Rule2 in figure 14 is probably caused since the model has been simulated using only 32 dimensions due to computational limitations and has been picked intentionally to illustrate the difference in performance despite a learning error. Using higher dimensions is expected to increase the accuracy of the model.

Table 4. Effect of learning history on Syntactic generalization

Experiment Mode	Number of Examples Shown	Generalization Accuracy*	Improvement in Accuracy*
Investigative Run	1	0.6 – 0.45 = 0.15	-
Exp. Sub-mode 2	2	0.65 – 0.38 = 0.27	(0.12/015)*100 = 80%
Exp. Sub-mode 3	3	0.77 – 0.4 = 0.37	(0.1/0.27)*100 = 37.037%
Exp. Sub-mode 4	4	0.8 – 0.42 = 0.38	(0.01/0.37)*100 = 2.702%

*Generalization accuracy is the difference between the semantic pointer representation of the correct answer (average of the two parts) and the largest incorrect answer in the answer output. Improvement in accuracy is calculated relative to the previous experiment mode by taking the difference between their generalization accuracy and expressing it as a % of the previous mode.

Earlier I thought that a minimum of two rules might be required for the model to be able to generalize an answer, but the model was able to generalize even when only one example rule was presented to it as shown in Figure 17. However, accuracy of generalization was very low.

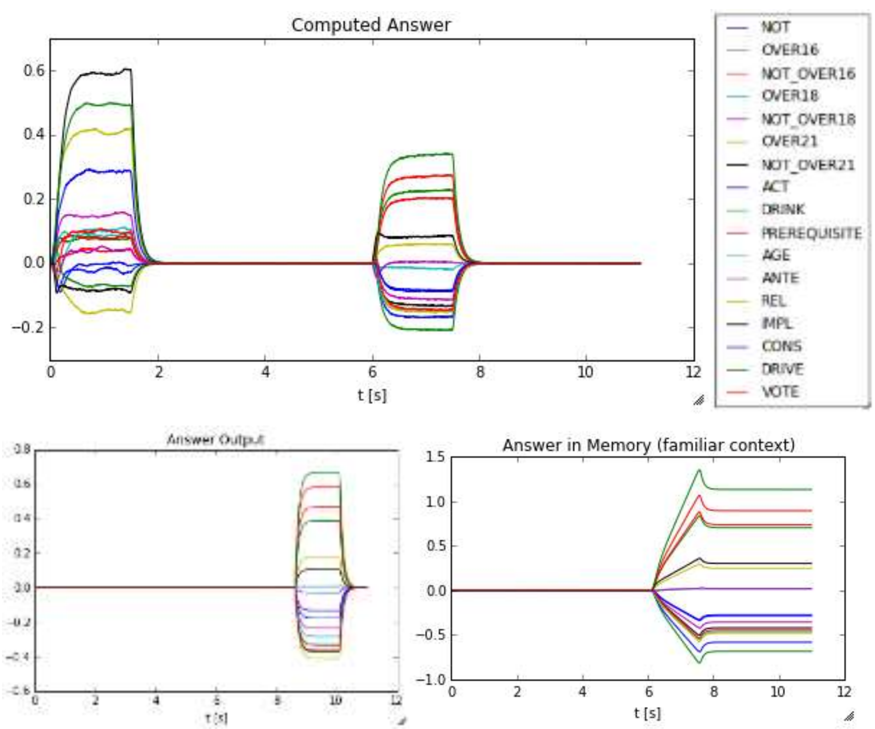

Figure 17. Results for the Sub-mode 4 in Experimental Mode2

Another interesting fact emerges from looking at table4 is that there is large % improvement between the investigative run (using only one example rule) and the Sub-mode2. There is also considerable % improvement in moving from Sub-mode2 to Sub-mode3. However, moving from Sub-mode3 to Sub-mode4 does not yield a very significant improvement. Thus the model improves rapidly with the first three examples and then improves less rapidly. This is similar to humans who are likely to provide more consistent responses after three examples and further examples might not make much difference to the accuracy of their response since they have already learned to generalize the concept.

3 Further Model Evaluation

The model evaluations for all the subsections in this section (except section 3.1 & 3.3) are done using the Sub-mode3 of the experimental mode2 (i.e., the generalization mode). The result for this sub-mode simulated with the default model parameters using LIF neurons as shown in figure18, has been used as a reference for these evaluations. The general accuracy for this mode using default parameters was found to be = 0.38 – 0.1 = 0.37. [Correct Ans: DRIVE+NOT_OVER16]

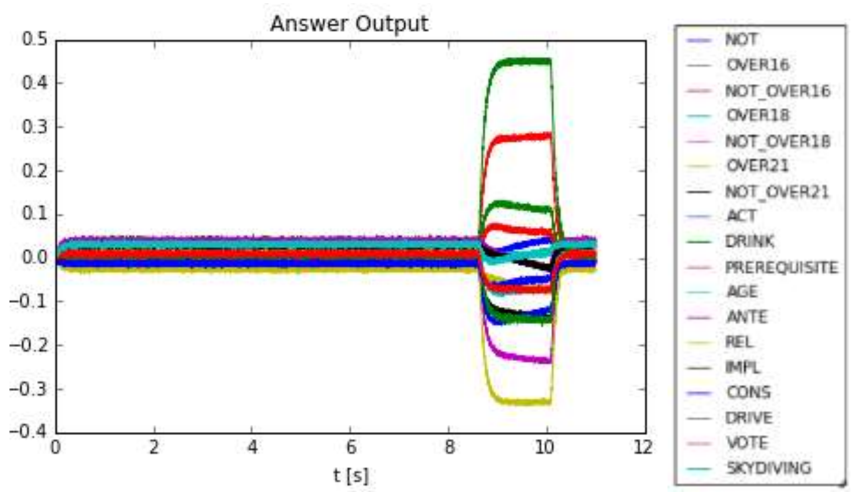

Figure 18 . Sub-mode3 Simulation result with default parameters using LIF neurons

3.1 Number of Dimensions

On increasing the number of dimensions, the generalization accuracy was found to improve (figure 19). This is because the semantic pointers which are being used for structured representations are generated randomly and can be similar to one another if lower dimensions are used. This means that two different concepts will be represented in very similar way and the model can get confused about them. I was able to simulate the model upto 48 Dimensions for Sub-mode2 (which took 2.5 hours due to computational limitations of my laptop) and could see a 48.148% increase in performance relative to the result in Sub-mode2 with 32 dimensions.

[Calculation: general accuracy = 0.81 – 0.41 = 0.4, % improvement in accuracy = [(0.4 – 0.27)/0.27]*100 = 48.148% , 0.27 is general accuracy with 32 dimensions taken from table4]

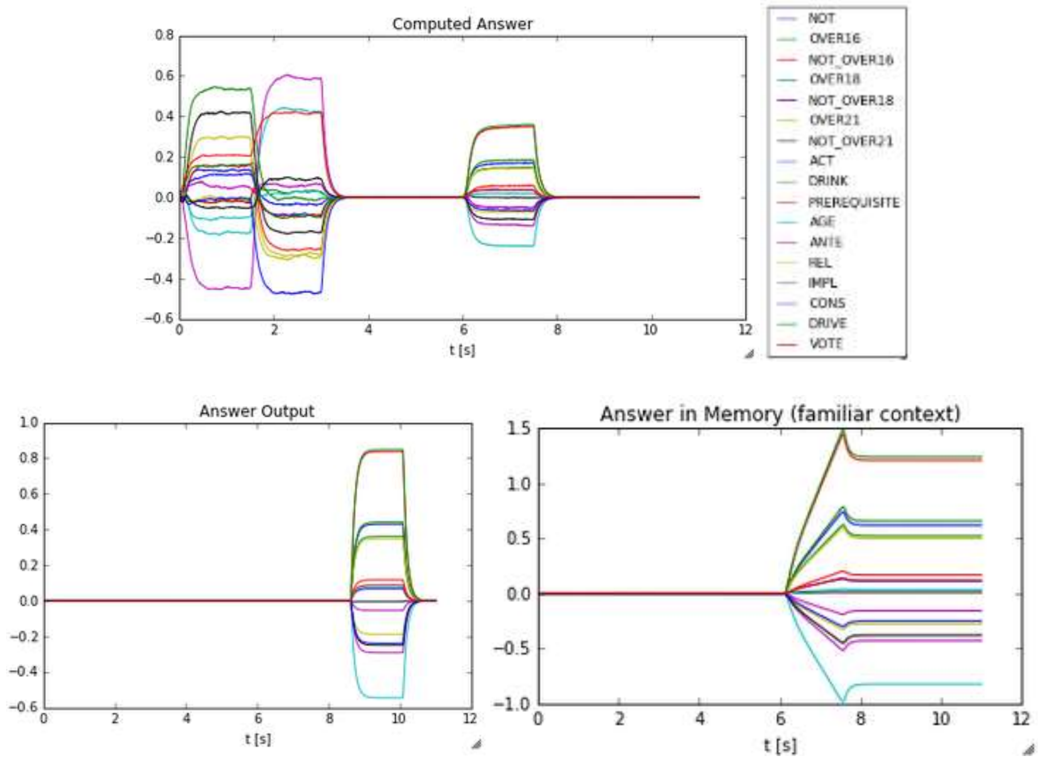

Figure 19. Results for the Sub-mode2 with 48 dimensions

I therefore predict that the performance of the model will further increase with increase in dimensions and then reach a saturation point beyond which no increase will be observed.

3.2 Degeneration of Cognitive Ability

Studies have shown that cognitive ability degenerates with age. Various neuroimaging studies have shown that while the brain undergoes significant structural change with age, the greatest declines are found within the frontal lobes. Regarding volumetric decline, both gray and white matter show significant decline across the lifespan, with the pre frontal cortex showing the greatest amount of loss and the steepest declines. Behavioral studies have also found age related decline in function to be greatest when tasks are dependent on frontal function. Thus the degeneration of the pre frontal cortex leads to deficits in cognitive performance across the lifespan [7].

Following three experiments were performed to see 1) how the model behaved on encountering diffused degeneration across all regions, 2) degeneration of only the learning populations in the right inferior frontal cortex and 3) degeneration of only the memory populations. All these experiments were performed with Sub-mode3 (three example rules) and using the same test rule as before (Correct Answer: Drive+Not_Over16).

Diffused degeneration of the brain

The number of dimensions were kept constant (32D) while the number of neurons across all the regions was decreased by 20% and then by 40% in another simulation. The original model was able to generalize accurately while the models with the degeneration gave erroneous results.

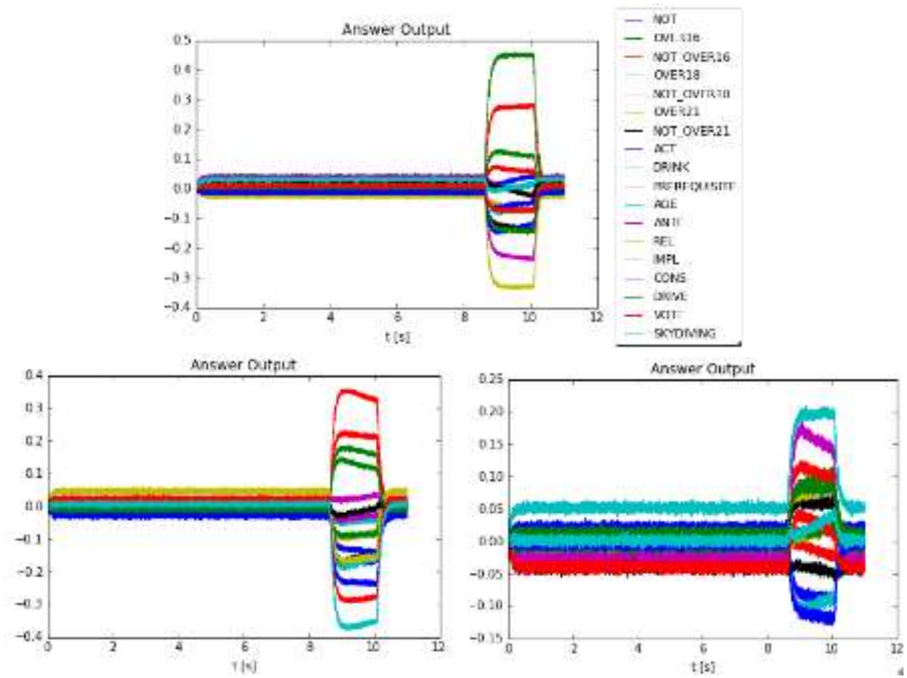

Figure 20. Output from the original model (top), model with 20% reduction in neurons (bottom left), and model with 40% reduction in neurons (bottom right)

Furthermore, the model with 20% degeneration was closer to the correct answer as compared to the one with 40% degeneration (which gave completely random answers) as shown in figure20. This result is similar to humans whose cognitive ability decreases with degeneration of the brain. 40% degeneration is a lot and therefore a person with that huge amount of degeneration would likely not be able to reason properly at all.

Degeneration of Right Inferior frontal Cortex

In the Wason model, reasoning happens in the prefrontal cortex, and it was found that reducing the number of neurons in the neural populations computing the transform lead to a decline in performance. The neurons in the learning populations were decreased from 320 to 256 (decreased by 20%), and it was observed that the generalization accuracy declined by 72.972%. [Calculation: generalization accuracy = 0.3-0.2 = 0.1, % decline in accuracy = [(0.37-0.1)/0.37 * 100] = 72.972%, decline is relative to the normal Sub-mode3 case]. However, on further decreasing the number of neurons to 192 (decline of 40%), the model gave erroneous result as shown in figure 21.

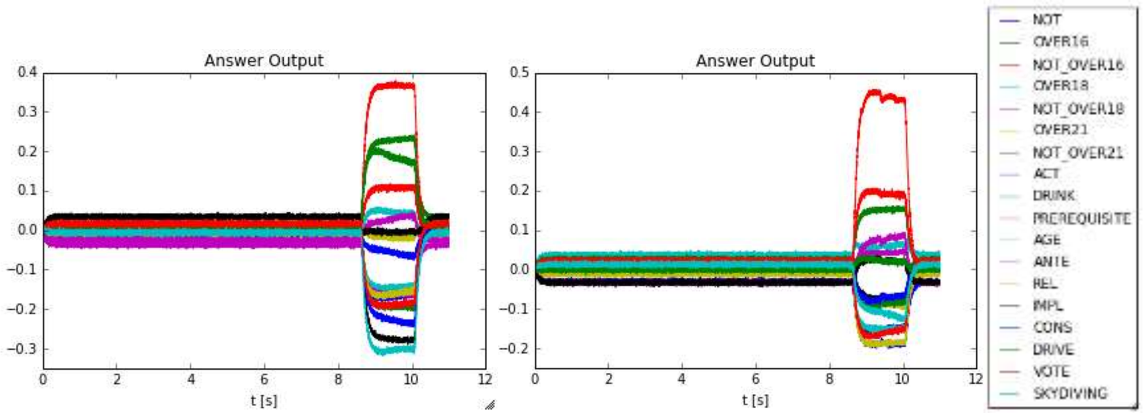

Figure 21. Output from the model with reduction in the neurons in learning populations – 20% reduction (left), and 40% reduction (right)

Degeneration of the Working Memory

The neurons in the memory populations were decreased from 1280 to 768 (decreased by 40%), and it was observed that the model was still able to provide the correct answer (figure 22), though the generalization accuracy declined by 62.162% [Calculation: generalization accuracy = 0.32-0.18 = 0.14, % decline in accuracy = [(0.37-0.14)/0.37 * 100] = 62.162%, decline is relative to the normal Sub-mode3 case].

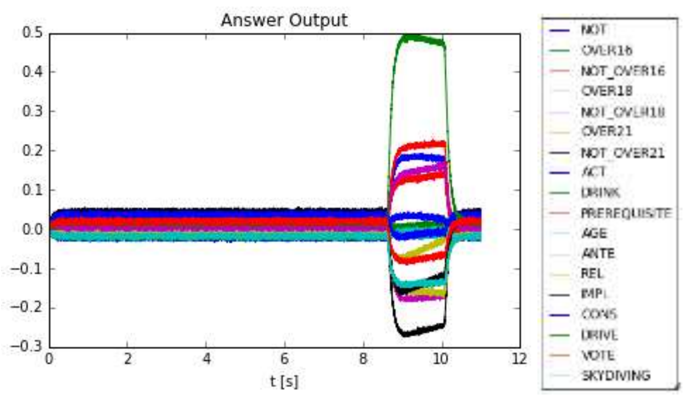

Figure 22. Output from the model with 40% reduction in the neurons in memory populations

Conclusion: The memory population is less susceptible to loss of neurons as compared to the learning population since the model was still able to generalize after 40% decline in neurons in the memory population, but gave erroneous results when the same was done to the learning population. However, the diffused degeneration of all the regions caused the greatest damage of the three cases. This is similar to what would be expected in a normal human brain since the overall degeneration of all the regions of the brain would have more severe effects than the degeneration of just one individual region by the same amount.

3.3 Performance on different set of sentences

The Learning and generalization experiments were also performed on a different set of sentences as shown below.

3.3.1 Learning

Abstract Rule: *If a card is yellow on one side, it should have an alphabet on the other side*
Correct Answer: YELLOW + ALPHABET
Familiar Rule: *If you are in grade9, you must be learning polynomials*
Correct Answer: GRADE9 + NOT_POLYNOMIALS

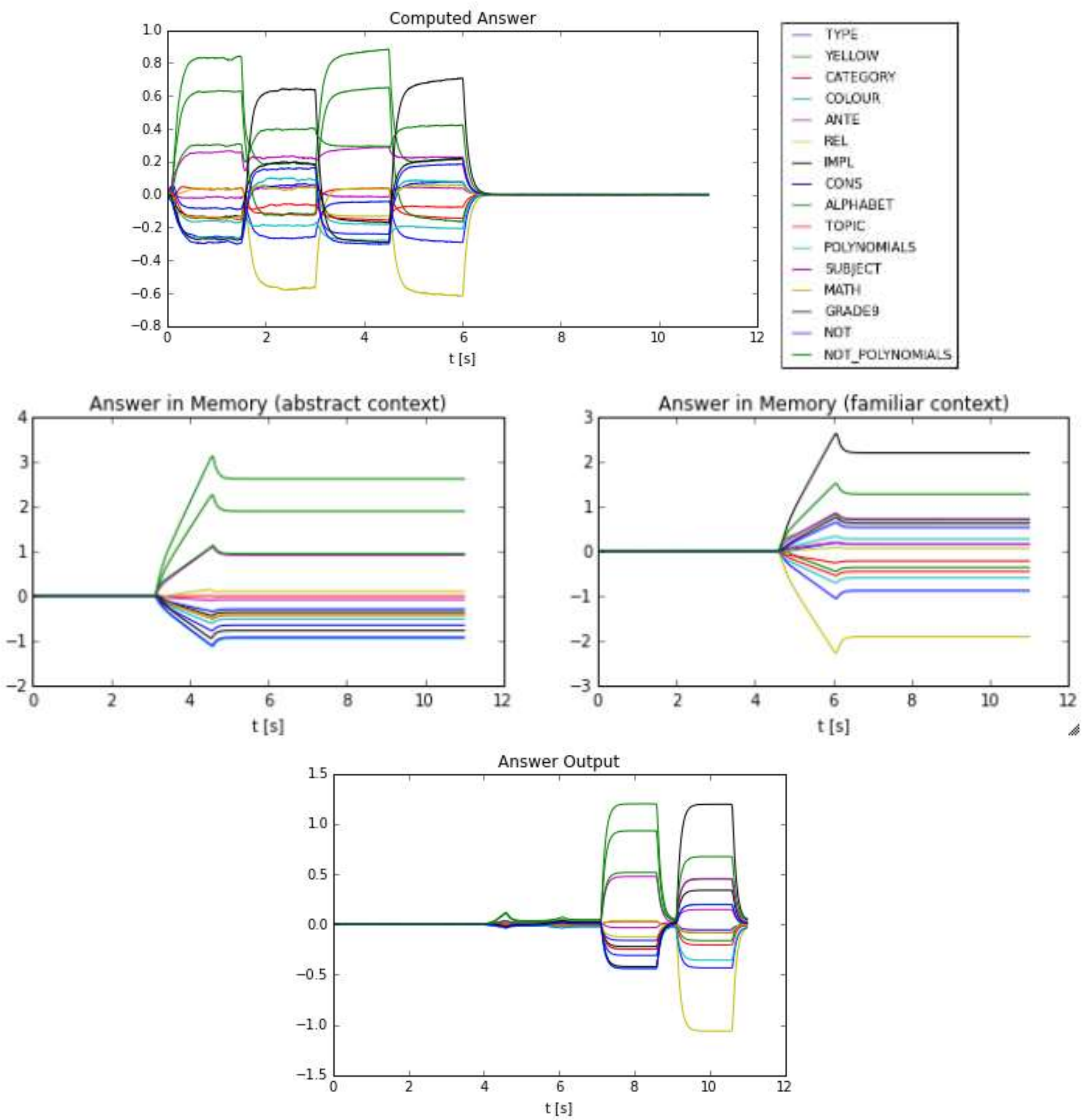

Figure 23. Results for the Experimental Mode1 for a different set of sentences

3.3.2 Syntactic Generalization

Rule1: *If you are in grade4, you must be learning division*
Correct Answer: GRADE4 + NOT_DIVISION
Rule2: *If you are in grade9, you must be learning polynomials*
Correct Answer: GRADE9 + NOT_POLYNOMIALS
Rule3 (Test Rule): *If you are grade12, you must be learning calculus*
Correct Answer: GRADE12 + NOT_CALCULUS
Rule4: *If you are in grade11, you must be learning functions*
Correct Answer: GRADE11 + NOT_FUNCTIONS

Rule1, 2, 4 were presented as examples to the model and the Rule3 was presented as a test rule. Figure 24 shows the results obtained. It can be seen that the model was successfully able to generalize to the correct answer. The overlap in semantic pointers in the computed answer when Rule2 and Rule4 are presented is due to lower dimensions (32D) being used due to computational limitations as mentioned before. The performance will improve on using higher dimensions.

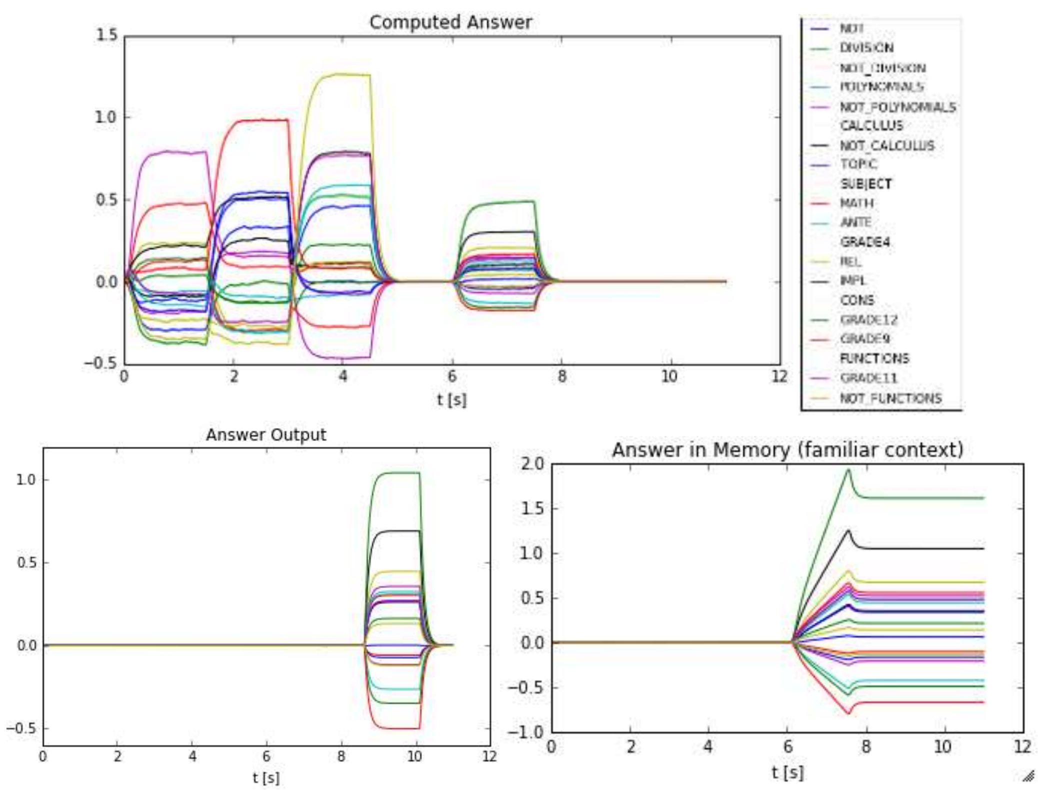

Figure 24. Result of Experimental Mode2 using a new set of example and test rules

3.4 Radius of neurons

As the radius of the neurons in the representation and learning populations was increased, the representation was found to degrade thus leading to the loss of accuracy of the answer. For example on doubling the value of the radius, the generalization accuracy was found to decline by 59.459% (figure 25). [Calculation: generalization accuracy = 0.35-0.2 = 0.15, % decline in accuracy = [(0.37-0.15)/0.37 * 100] = 59.459%, decline is relative to the normal Sub-mode3 case].

This is because the radius is indicative of the amount of information which the neurons would represent. Hence as the radius is increased, the same number of neurons would need to represent more information and thus the answer keeps becoming more and more inaccurate. Increasing the radius implies that the neurons will have a smaller change in firing rate for a given change in the stimulus so their representation accuracy decreases.

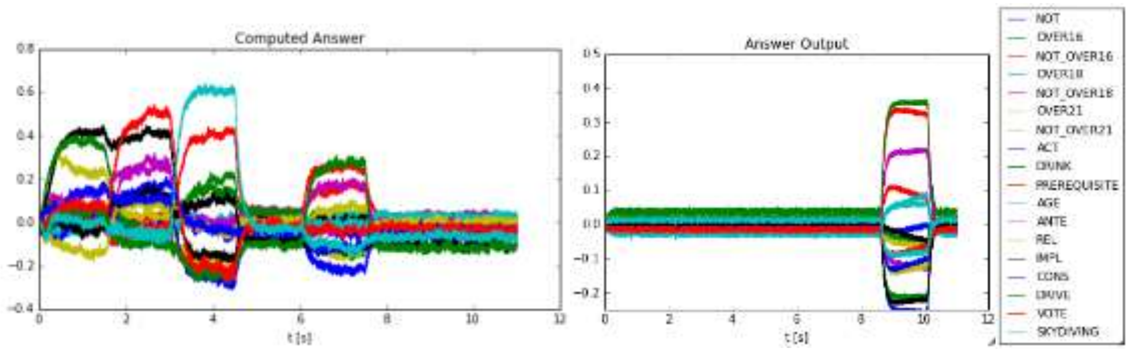

Figure 25. Answer computed by the model (left) and answer output by the model (right) on doubling the value of the radius of neurons (radius = 2)

3.5 Max firing rates of neurons

When the max firing rates of the representation and learning populations were decreased from a range of (200,400) to a range of (100,200), the result shown in figure26 was obtained.

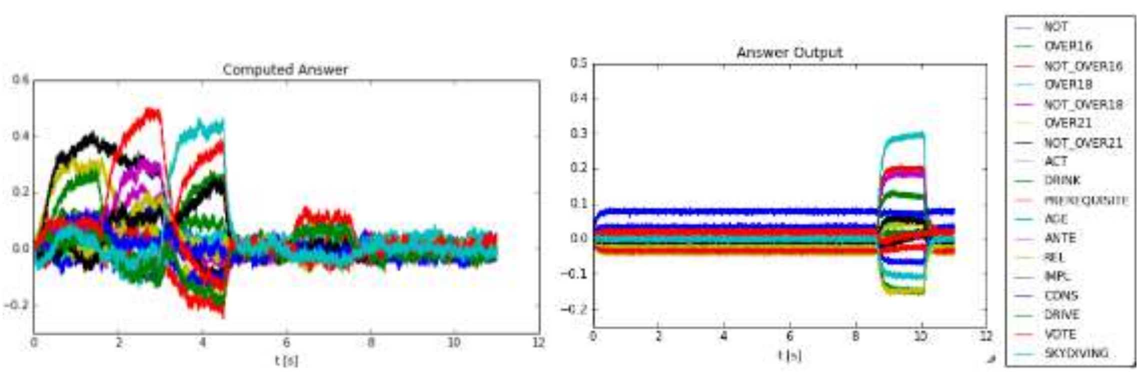

Figure 26. Answer computed by the model (left) and answer output by the model (right) on doubling the value of the radius of neurons (radius = 2)

Thus the firing rates had a huge impact on the representational accuracy thus making the model not work at all. This is because lower max firing rate implies that the neurons will saturate earlier and thus would not be able to represent the stimulus accurately.

5 Future Improvements

The context determination in the model has been done using a similarity measure which also takes the number of example rules shown to the model into consideration. However, more work can be done in order to investigate more natural ways of determining context.

Another concern is that due to computational limitations, the model hasn't been tested using large number of simulations, and the inferences mentioned in the report are based on relatively small number of simulation trials. Thus, it would be beneficial to test the model using better computational resources and confirm these inferences to make sure that they are not subject to confirmation bias.

Additionally, provided computational capability, it would be nice to determine the upper-bound on the number of dimensions beyond which the performance of the model would not increase anymore.

6 References

[1] Eliasmith, C., & Anderson, C. H. (2003). Neural engineering: Computation, representation and dynamics in neurobiological systems. Cambridge, MA: MIT Press.

[2] Eliasmith, C. (2005). Cognition with neurons: A large-scale, biologically realistic model of the Wason task. Cognitive Science Society, Stresa Italy.

[3] Eliasmith, C. (2013). How to build a brain: A neural architecture for biological cognition. New York: Oxford University Press.

[4] Physiology Web: Neuronal Action Potential –Refractory Periods. Available Online: http://www.physiologyweb.com/lecture_notes/neuronal_action_potential/neuronal_action_potential_refractory_periods.html

[5] The Stanford NeuroBlog: What Is The Synaptic Firing Rate Of The Human Brain? Available Online: http://neuroblog.stanford.edu/?p=4541

[6] Simultaneous unsupervised and supervised learning of cognitive functions in biologically plausible spiking neural networks. Center for Theoretical Neuroscience, University of Waterloo. Available Online: http://mindmodeling.org/cogsci2013/papers/0058/paper0058.pdf

[7] Frontal lobes and aging: Deterioration and Compensation, Roberto Cabeza & Nancy A. Dennis.AvailableOnline:http://canlab.psych.psu.edu/docs/Publications/Chapters/CompensationChapter_StussKnightBook_CabezaDennis_final.pdf