Application of Artificial Neural Networks for modelling cognitive dimensions

Pedro Ferreira da Costa
pedrohfcosta@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

October 2018

Abstract

The relationship between the brain and cognition remains unclear, despite several decades of functional neuroimaging research. One limitation is that the cognitive processes we attempt to match to brain activity are taken from psychological constructs derived in a somewhat ad hoc manner. This project took a different approach, taking advantage of developments with artificial neural networks (ANNs) to learn shared mechanisms. The purpose was to evaluate the execution mechanisms between multiple cognitive tasks without relying on predefined cognitive domains. Therefore, a Recurrent Neural Network was developed to perform six cognitive tasks with an accuracy of 93%, that tapped on the processes of reaction, inhibition and working memory. With regard to the obtained model, it was tested if the mechanism provides a good explanation for the activation patterns in the brain regions previously associated to the cognitive processes. Although comparisons between the models activations and real brain data bared little similarities, the models mechanisms expressed an effective system of interpreting each task. It was clear, by analysis of the models interpretation of the input dataset, that the concept of task and moment of reaction were important factors for the correct solution. From the study of node variation along trials, one stood out (unit 28) by displaying a behaviour similar to inhibition control in biological systems. This work intended to provide novel insights into both brain and cognition, suggesting a potential parallelism between the artificial model and the biological processes. This perspective can contribute to a clearer interpretation of the cognitive processes.
Keywords: Cognitive Pathways, Ontologies, Machine Learning, Cognitive Processes

1. Introduction

For millennia scholars have wondered what makes humans be able to apply logic, speak, think and think about thinking, traits that seem absent or greatly reduced in other animals. These questions are at the core of humanitys insatiable curiosity and their complexity has led to different answers throughout the ages. The dissemination of the scientific methods and the evolution of technology has allowed different fields such as psychology and neuroscience to study some of the mysteries of the brain but with limited success on the overall mechanisms that result in intelligence. Cognitive neuroscience, a research field at the intersection of the two, concerns the study of the neural processes and mechanisms that give rise to thought and understanding. During everyday life, humans perform complex behaviours, such as memorising names or planning the dinner for the family. This behaviour generally depends on multiple overlapping faculties and usually allows for several solution strategies. As an example, the simple task of crossing the street requires motor dexterity from the legs, requires attention mechanisms to observe when the pedestrian light turns green and ultimately requires memory skills on interpretation that the green light means it is safe to move. To overcome this challenge in the study of human cognition, over the last 150 years, psychology has developed cognitive tasks aimed at isolating a given cognitive ability (e.g., sustained attention, working memory, response inhibition). With these efforts a broad cognitive ontology was developed and is still used in cognitive neuroscience to date. It is a formal representation of the types of entities in the cognitive domain and the relationship between each other. However, this cognitive ontology has been arrived at in a relatively ad hoc way and does not take into consideration the functional organization of the human brain. This hinders the process of understanding the neural mechanisms and prevents a full mapping between functional and anatomical structures [13, 3].

With the recent surge in applications of artificial neural networks to solve cognitive tasks, there is an increasing amount of literature that highlights similarities in biological and artificial structures. Ma-

chine Learning might prove to be the conductive thread between current ontologies (i.e. based on cognitive paradigms) and a more structural (i.e. focus on networks activations) approach to cognitive ontologies by effectively mapping one to the other [12]. Contrarily to brain systems, artificial neural networks allow full access to its inner calculations. Even though the interpretation of how each component contributes to the general outcome remains challenging, the understanding of its mechanisms might provide an alternative way to formulate a cognitive ontology and shed light on the relationship between cognitive paradigms and the structures that originate function. Based on recent developments from deep learning (e.g., Googles DeepMind [14, 7]), this work has attempted to use artificial neural network approaches to learn the complex relationships in cognitive tasks. Such artificial neural networks have previously been shown to be able to simultaneously capture multiple cognitive tasks in a single model [2]. These prior studies indicate that the proposed work was plausible and forms the starting point for the translation into more complex cognitive tasks thought to be related to frontoparietal networks.

This project set out to build and train a neural network to solve six classical cognitive tasks in order to better understand the mechanisms intrinsically created by the algorithm for addressing each cognitive process. This allowed modelling the compositionality and clustering of task representations [14, 8], which allowed us to understand from a bottom-up, purely mechanism-driven way how different tasks group together. This project was designed as a first stepping-stone towards a larger goal of creating a mechanistically based cognitive ontology, using neural networks as the medium for an effective mapping between cognitive paradigms and its representations without relying on predefined cognitive processes. This approach has the potential for providing a radically different understanding of the relationship between brain networks and cognition.

2. Methods

2.1. The Dataset

For the purpose of training and testing the model, a dataset was generated with 2000 trials covering 6 tasks that the model was intended to learn: The Go Task, the Anti Go Task, the Memory Go Task, the Memory Anti Go Task, the Reaction Go Task, and the Reaction Anti Go Task. The aforementioned were based on tasks described in the work of Yang et al. [14].

Each trial was composed of two separate matrices - the input matrix, which was fed to the model, and the output matrix, which was employed to evaluate the models prediction and define the error function. Independently of the task of the respective trial, all input matrices have the dimension 71 by 600 and all output matrices have the dimension 33 by 600. The matrices lines represent nodes and the columns represent time-steps. If each time-step is considered to last 10ms, then each trial has a reasonable duration of 6 seconds. The input signal (71 x 600) can be decomposed into 4 different aspects the rule signal, the fixation unit and the two modalities. The rule signal corresponds to the first 6 nodes and is implemented as a one-hot vector (in which one value is expected to be 1 and the rest to be 0) to represent which one of the six tasks is being executed in a given trial. The rule signal is constant across every time-step of a trial. The 7th node represents the fixation unit a binary value that cues when and whether the model should react to the stimulus or not. Across time-steps, if the value of the unit is 1, the model should not react and should maintain fixation. When the value drops to 0, this signals that the model should react to the stimulus according to the task (i.e., consistent with the rule representation). The latter 64 nodes represent two separate modalities of 32 nodes each. Each modality can display an arbitrary signal, by activating one of its 32 nodes. In the original paper, from where this projects tasks are modelled, they proxy rodent behavioural tasks, with the inputs representing a direction in a circle that a rodent should take in order to collect its reward [14]. In a similar way, the signal can represent the direction to which the behavioural response should be executed. It is in modality one or modality two that the stimulus, to which the model should react, is represented. There is only one stimulus per trial and its duration depends on the task in hand. It is important to note that the need for both modalities instead of just one arises from the next stages of the project in which new tasks will require the model to discriminate between stimuli in each modality. In order to maintain the dataset structure, it was defined to already account for these new tasks. To simulate noise in underlying neural representations, the input matrix is combined with additive Gaussian noise (i.e. $N(0;0.1)$). In total there are $N_{in} = 1 + 6 + 32 * 2 = 71$ input units.

The output signal (33 x 600) is a one-hot vector representation of the desired action of the model. It is divisible into two components the fixation unit and a 32-unit reaction output. As with the inputs fixation unit, the outputs fixation signal sets a unitary value for when the model stays put (fixates) and drops to 0 to react in a given direction. In a computational sense, it allows the model to have an output signal even when it is not required to respond. The reaction output is structured similarly to the inputs modalities and it corresponds to the input representation where the model should define

the response to the stimulus when the fixation is dropped. In total there are $_{out} = 1 + 32 = 33$ output units.

Figure 1 depicts an example of a Go Task trial that serves as both input and output for the ANN. 1.a refers to the Rule signal which indicates which of the 6 trials is being considered, in this case unit 0 (i.e. Go task) is on throughout the trial. 1.b indicates the input of the fixation unit. It falls to zero when the model is expected to react. 1.c signals the expected output of the fixation unit. The model is expected to mimic the fixation unit, in order to understand when it should react. 1.d refers to the input modalities, which consist, each one, of a 32-vector input per time step that simulates the direction of the stimulus. For this project, only one modality has a signal for each trial. 1.e represents the expected output signal, the direction the model should take for a given trial, depending on the task. In this case, the model is expected to follow the stimulus in modality 1, when the fixation reaches zero.

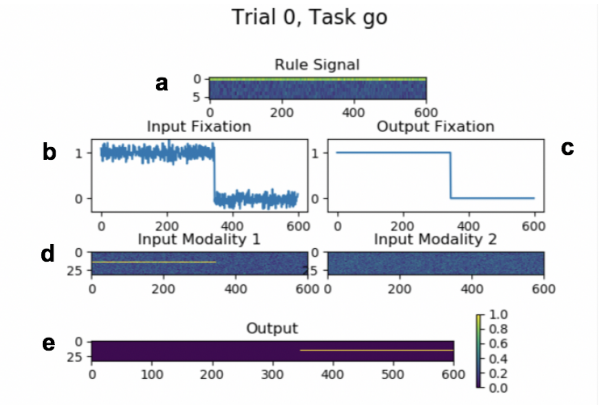

Figure 1: Representation of the dataset provided to the model for a trial of a Go Task.

2.2. The Tasks

In choosing the learning tasks for the model, three aspects were taken into consideration: each tasks commonality across neurophysiological studies, its simplicity for posterior analysis and the flexibility to allow the same input dimensions for different tasks. Based on the work by Yang et al. [14], 6 tasks were chosen and implemented that included variants of saccade, inhibitory control, parametric working memory, memory-guided response and reaction time.

All tasks have the same input and output dimensions, as described in Section 2.1 so what differentiates between tasks is the rule signal and how the stimuli are expressed across the modalities of the input and the expected reaction to the stimulus in the output. For each task only one dedicated unit could be active in the rule signal, as was previously explained. In order to assure reproducibility and to facilitate posterior analysis, the random values in each trial are registered and saved (these include the duration of the stimulus, the time of reaction or the task being covered by the trial).

The **Go Task**, which could be described with the instructions: react in the direction of the stimulus when the fixation stops, is a fundamental common across many behaviour tasks [4]. In the present dataset, the stimulus is randomly assigned an input representation in one of the two modalities. When the stimulus disappears, the fixation stops, and the model is supposed to react in the direction of the stimulus (one of the 32 units).

The **Anti Task** can be described as: react in the opposite direction of the stimulus when the fixation stops. This involves an inhibitory reaction from the participant, not directing their gaze in the direction of the stimulus [9]. The task is the inverse of the Go Task as the expected reaction should be taken to the opposite direction of the original stimulus.

The **Delay Go Task** can be described with the instructions: memorize the direction of the stimulus and react to it when the fixation drops [5].This type of cognitive task would only be solvable with a model with some form of memory of past events, such as the LTSM recurrent network presented here (see Section 2.3).

The **Delay Anti Task** is the inverse of the previous task, being described as: memorize the direction of the stimulus and react in its opposite direction. Unlike the previous task when the fixation stops, the model should react in the opposite direction to the memorized stimulus. The **Reaction Time Go Task** is similar to the Go Task but the reaction to the stimulus should be taken as fast as possible, while the fixation never drops, the moment the stimulus is detected.

The **Reaction Time Anti Task** can be described as: react in the opposite direction of the stimulus, as fast as possible; as with the React-Go task, the fixation signal never drops to 0.

2.3. The Model

The following image is a pictorial representation of the model that was developed for this project with the goal of solving 6 cognitive tasks with temporal dependencies.

After building the dataset, the deep Recurrent Neural Network (RNN) structure was defined and implemented using Keras , a high-level neural networks API with TensorFlow as its computational backend. The basic recurrent unit applied was an LSTM, as it is able to maintain information for longer periods than regular recurrent units and store a cell state, simulating the networks memory storage. Overall, the software architecture involved

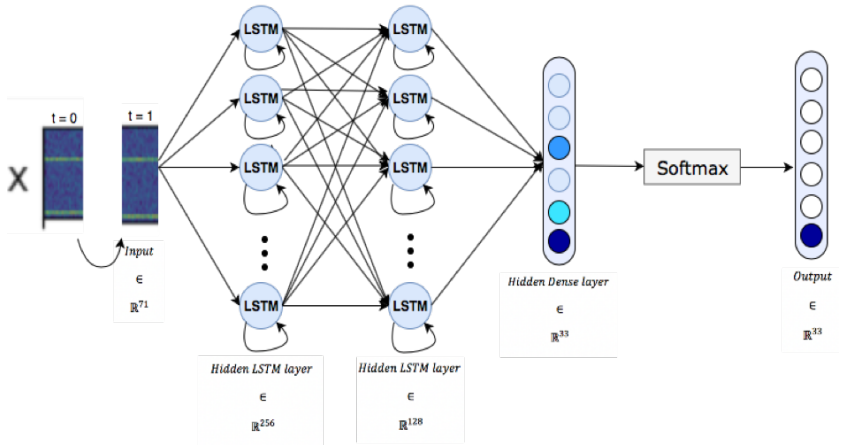

Figure 2: The proposed neural network architecture. The model is composed by two densely connected LSTM layers and a third dense layer that, after a softmax activation outputs a vector of size 33.

the use of 4 specific Python packages (i.e. keras, numpy , sklearn , and matplotlib). The entire model was trained end-to-end from the dataset created for the project. Most of the models hyper-parameters were chosen using empirical heuristic methodology. The model was trained with a first layer of 256 LSTM units, a second layer of 128 LSTM units and a third dense layer with a softmax as the activation function.

The model was fully interconnected between layers 1, 2 and 3 with 537,249 weights, all of which were trained using a categorical cross-entropy loss function (Equation 1), where q is the approximate distribution and p is the real distribution. The optimization function corresponds to the minimization of the sum of the loss functions across every time-step j and trial i of a given batch. Each batch was composed of 64 randomly chosen trials Equation 2.

$$H(p,q) = -\sum_{n=x} p(x)log(q(x)) \tag{1}$$

$$Minimize \sum_{n=i}\sum_{n=j} H(p_{i,j}, q_{i,j}) \tag{2}$$

In order to effectively train the model, the Adam optimizer method was used, a type of stochastic gradient descent, with the parameters following those proposed by the seminal paper of Kingma and Ba [6]. The learning rate used was 0.001 with a decay rate for the 1st and 2nd moments of 0.9 and 0.999 respectively.

2.4. Model Analysis

After successfully implementing the model to correctly learn the six cognitive tasks, it was this projects intention to understand the mechanisms used by the model to correctly solve the tasks, based solely on the input it is given. Towards this goal, a number of analyses on the models activations, parameters and unit variance were run on Python.

Understanding how the activation for each layer varied according to the different variables present in the dataset (i.e. task, moment of reaction, delay period, stimulus duration) is a multidimensional problem. Therefore, t-distributed stochastic neighbor embedding (t-SNE), an algorithm that allows representation of high-dimensionality data in two dimensions [11], was applied to the model while solving the testing set. A variance study was also conducted to study the importance of each unit for the correct execution of different functions required in each task.

3. Results

The first experimental objective was to successfully train a model to perform 6 different cognitive tasks. This objective was achieved with a general accuracy of 93%. The dataset on which the models were trained contained 2000 trials, divided into a training set of 1600 trials (i.e. 80% of the full dataset), a validation set of 200 trials assessed at the end of each epoch, and a testing set containing 200 trials, to define how the model generalizes for values not seen before.

The accuracies for each of the tasks are reported on Table 1. Comparatively, the model underperforms on the Reaction Time tasks. It is also notable how, for the other 4 tasks, the accuracies on the 200 trials are alike, with special emphasis on the Go and Anti tasks (i.e. 85.9% and 85.8% respectively).

3.1. Model Analysis

To characterise how each layer contributed to the model performance, the different layers activations were analysed with the non-linear dimensional-

	Accuracy					
	Go Task	**Delay Go Task**	**Reaction Time Go Task**	**Anti Task**	**Delay Anti Task**	**Reaction Time Anti Task**
Model 5	85.9	86.3	61.3	85.8	87.1	71

Table 1: Model 5 accuracy for the different tasks learned.

ity reduction technique, t-SNE [11].The trials are colour labelled by the different variables that the model was hypothesized to have to learn for correct performance (i.e. the task, time the model should react, delay period between stimulus and reaction and duration of the stimulus). Through this method it is possible to analyse both the final outcome and the activations of the inner layers, including the cell states of the LSTM units. It is evident from the Figure 3.a, that there is a significant similarity between trials of a given task across the cell states of the first layer of the LSTM. The 6 clusters are grouped by the task being performed, which demonstrates how the model represents how different tasks relate to different expected reactions. It is also evident from Figure 3.b that trials are organised by when the model has to react, with later responses occupying the distal region of each cluster and the earlier ones being represented in the proximal regions. This is consistent with the cell state of the first layer which allows the information to be retained with minimal losses between time steps, allowing for its later retrieval.
The t-SNE applied to the output activation patterns of the first LSTM layer (Figure 3.c and Figure 3.d) highlights a similar result, which is expected as the layer is directly influenced by the values of the cell-state. Despite this, there is no clear distinction between task representations, with some tasks appearing more overlapping (i.e., Go and Delay Go tasks and the Anti and the Delay Anti tasks). Furthermore, there is an apparent symmetry between both Go/Anti tasks and Delay Go and Delay Anti tasks, an interesting result as the tasks behave symmetrically (i.e. the Go task should react towards the stimulus and the Anti task should react against it). Both Reaction Time Go and Reaction Time Anti tasks are representationally distinct from each former, an outcome present throughout Section 3 and that will be addressed in the Section 4. Figure 3.d delineates a clearer representation of a gradient suggesting when the model intends to react.

To help us better understand the importance of each unit for the correct execution of a given task, the variance of each models units was studied across trials for a given task. Each trial is composed of 600 time-steps with varying time of reaction and stimulus period. The moment chosen for the variance analysis was locked to 10 time-steps after the model was expected to react (Tgo); this allowed for the activations to settle into the correct response signal. High intra-task variance is expected to reveal the units responsible for functions such as the moment of reaction or the direction of the reaction (i.e. variables that change between trials of the same task). The results obtained are depicted in Figure 4.

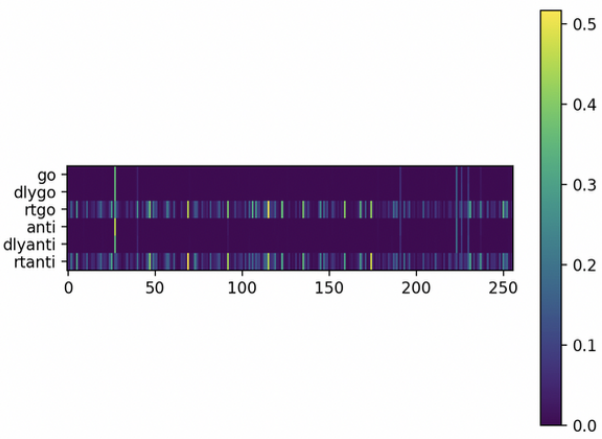

Figure 4: Variance of each first-layer unit across each task

The Reaction-Time and the Anti Reaction-Time tasks display a higher variance between trials along the units of its first LSTM layer. This finding agrees with the t-SNE representation that displayed the two tasks further from the rest. Due to the high variance present in both Reaction Time tasks, the lower variance for the units in other tasks are not visible. It is apparent that some units (e.g. units 28, 224, 227 and 231), display a function that is common across every task.
Subsequently, the variance of the node's activations across every trial is compared, independently of its task, to the variance between averages of the task specific trials. This allows us to quantify how each node relates to the desired outcome, as a node with a greater variance across trials has a more significant role on the final output than one with little variation. Also, by observing how the variance changes between different tasks, it is possible to discover whether a specific node is responsible for a function common to every task (the variance across task-averaged trials would be low), or alternately, for a function that is directly responsible for different aspects of a specific tasks (higher variance between task-averaged trials).

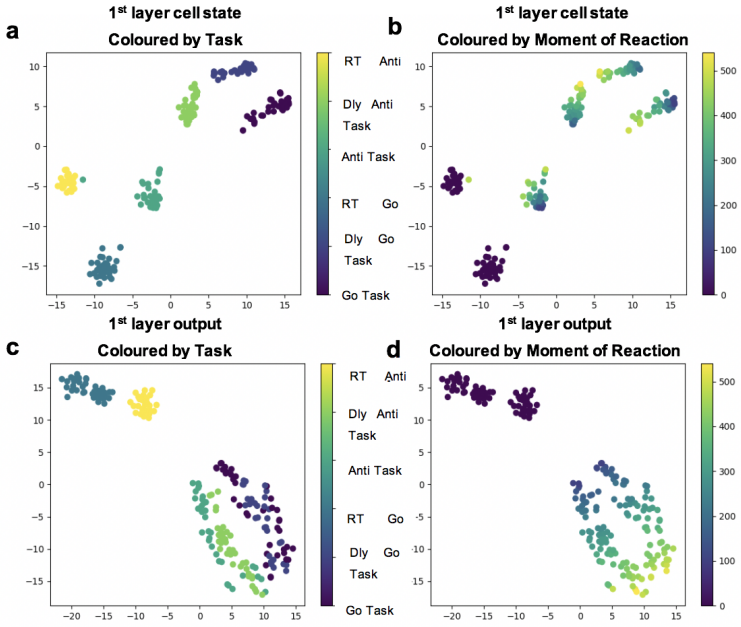

Figure 3: Four t-SNEs plots, each representing 200 trials as scattered points relating to Model 5s first layer. a: cell state activations across time with the colour label of the respective task. b: cell state activations across time with the colour label of the moment of reaction (Tgo). c: LSTM cell output across time with the colour label of the respective task. d: LSTM cell output across time with the colour label of the moment of reaction (Tgo).

For the first layer of the model, although most of the nodes display little variance, some display a clear variation of the value outputted from the first layer of LSTMs to the second layer. The most noteworthy ones are, in descending order, units 28, 160 and 93 with variance of $\sigma^2 = 0.466$ $\sigma^2 = 0.427$ and $\sigma^2 = 0.424$ respectively. As expected, the general variance of the nodes decreases when the comparison between 6 examples (one for each task averaged across trials) are considered. There is also an abrupt decrease in variability of unit 28 from $\sigma^2 = 0.466$ (the highest in the first analysis) to $\sigma^2 = 0.142$ significantly lower than the variance for unit 93 and 160 $\sigma^2 = 0.282$ and $\sigma^2 = 0.312$ respectively. It is reasonable then, to consider that unit 28s function is related to something common to every task.

3.1.1 Removal of unit 28 from the first layer

Unit 28 from the first layer of the model displayed interesting patterns of activations. The variance displayed across trial activations exhibited a significant magnitude between tasks, with special emphasis on the Go, Delay Go, Anti and Delay Anti tasks. To further understand its effects on the overall output of the model, the unit was eliminated by reducing its weights to zero. The model was subsequently analysed on the different tasks and the accuracy is displayed in Table 2

It is evident a clear decrease in overall accuracy from 93% to 49%. This drop is supported by a decrease of accuracy in all tasks but the Reaction Time Go and Reaction Time Anti tasks. These two have nearly reported the same percentage of accuracy with and without the 28th unit from the first layer. The largest drop in accuracy is verified in the Delay Go task (i.e. from 86.3% to 36.9%). The model is composed of 384 units so the impact of a single unit in the overall accuracy is remarkable.

With the goal of further analysing the effects of the elimination of unit 28, the model outputs were compared to the original model with special focus on the tasks where accuracy dropped heavily.

The analysis made clear that without unit 28 the model was not able to follow fixation before the expected reaction. The altered model reacts to the stimulus the moment it is shown, resulting in a lower accuracy in the Go, Anti and Delay tasks and, logically, not affecting the Reaction Time tasks as they do not depend on the fixation input.

In summary, for the first LSTM layer, unit 28 displays importance across every trial that requires to follow fixation, which is not the case for units 93 and 160 which appear to be only relevant for the correct solution of the Reaction Time tasks. These

	Accuracy (%)					
	Go Task	**Delay Go Task**	**Reaction Time Go Task**	**Anti Task**	**Delay Anti Task**	**Reaction Time Anti Task**
Model without unit 28	46.1	36.9	60	44.1	41.4	71.2

Table 2: Accuracy for the model with unit 28 from the first layer removed on the different tasks learned.

units have relatively transparent functional roles, whereas other nodes are also functionally involved despite not varying their activations a lot between trials. This would be the case for units responsible for functions common to every trial (e.g. related to the fixation unit; regard stimulus direction). The second LSTM layer presents more significance for the desired output as expected, although most node variance seems to be accounted for intra-task purposes as it decreases to close to 0 when only variance between tasks is accounted for.

4. Discussion

4.1. Interpreting the model

By comparing different model structures, this project was able to find a model architecture that successfully generalized for new examples without overfitting the training data (i.e. low bias and variance). The model accounted for a high number of trainable parameters (i.e. 537,249), mainly due to the number of variables between trials of the same task, such as stimulus direction and moment of reaction. This also allowed the model to obtain low variance without using regularization methods. Despite this, the high number of weights and units made the model analysis difficult.

The study of trial representations on both layers revealed the expected outcome the first layers activations displayed a clear task clustering and a gradient variation on the moment of reaction while the second layers activations across trials were more dissimilar and did not appear to cluster significantly. The variance of unit activations across trials is also more dominant on the second layer than on the first. Both results are justified by the second layer activations depending directly from the output of the first layer with each unit connected to every unit from the previous layer.

Due to the previous outcomes and the first layer being connected directly to the models input, it is possible to assert that the first layer allows for a clearer interpretability of each units function. The first layer appears to display an objective representation of the different modelled variables, with its units representing clearer roles in the overall solution (as is the case of unit 28). The second layers interpretation is more challenging as the variables do not match the layers representation and the different processes result from the influence from different units, which also contribute to more than one function. Therefore, the first step to fully understand the underlying computational mechanisms of how the model captures each type of task is to examine the first layer.

These mechanisms display a resemblance to the brains functionality representation where the somatosensory periphery of the brain allows for a one-to-one mapping between regions and basic functions, but central areas are more difficult to interpret as functions and regions display many-to-many mappings and a single structure might not relate to an objective function [10].

4.2. The curious case of the Reaction Time tasks

Out of the six cognitive tasks the model learned, there was a clear distinction between how the model processed the RT (Reaction Time) tasks when compared to other tasks. Analyses on the trial representation using the t-SNE algorithm for the first layer revealed that trials clustered according to the tasks and that the activations for trials of RT tasks greatly differed from the other tasks both on the cell state and on the layers output. Besides this differentiation, the variance analysis concluded that RT tasks activations for the first layer varied greatly across the 256 units relatively to the other tasks. This allows the assumption that the model has greater ease on solving the four other tasks, as the trials in a given task do not change their activations by a large margin to successfully account for the intra-task variability (i.e. moment it should react, stimulus direction). This assumption is supported by the models accuracy which is 20% lower for the RT tasks than for the other tasks.

There are ultimately two differences between the RT tasks and the other four. These are the necessity to respond from the first iteration without any interval to interpret the stimulus and the requirement to ignore the fixation input which stays on for the whole trial and, contrarily to other tasks, does not convey the moment to react. Both differences should account for the change in representation. This work established the role of unit 28 for the trailing of the fixation input. Therefore, it can be assumed that this unit and other units that contribute to the execution of the fixation input mostly need to be ignored, resulting in an essentially different activation map.

Despite the higher variability present in the second layer of the model for every task, both RT tasks present high variance for a larger number of units but lower variance in some specific units (e.g. unit 20). These results are senseful as the first layer, which depicts higher variance across units, is densely connected to the second layer resulting in variance transmission. It is possible that the units with lower variance are a result of the model learning to deactivate units that are responsible for functions not involved in the RT tasks.
In conclusion, as the RT tasks are currently modelled, they present little relation to other tasks, which does not resonate with the brains interpretation of RT tasks. They also appear to be unreasonably complicated for the models execution, resulting in the need to recruit a large number of units and a lower accuracy rate. A different approach to the modelling of the task should be considered in the future, which might result in a simpler model that as the other four tasks might require a lower number of units to represent successfully the appropriate outcome.

4.3. The functionality of unit 28

Out of all the units analysed, unit 28 from the first layer stood out due to its high variance across non-RT tasks. In fact, when studying the variance across all trials and doing the same study without regarding RT tasks, unit 28 was one of the few whose variance remained virtually unchanged. It was possible to assert that its function related directly to a process not required by the RT tasks. Furthermore, the decrease in variance when studying the values averaged across each of the tasks, allows the assumption that unit 28s function regards a process similar to every task, as the averages of variances reported similar results. In order to better assess unit 28s functionality and understand how it might influence the general outcome of the model, its weights were set to 0 (effectively lesioning it). Its importance for the models success was immediately asserted, as the overall accuracy dropped to 49%. In comparison, the removal of another random unit resulted in a decrease of only 2% when compared to the original model. By assessing how the model fared in each task separately, in became clear that the performance on the RT tasks remained virtually unchanged. In contrast, every other tasks performance was heavily diminished by the units absence with the Delay Go task decreasing its accuracy by 49.4%. The output of the model revealed the direct impact of the units removal the model failed consistently to follow fixation and regarded the stimulus from the moment it was presented. As the Go and Delay tasks success depends on the model only reacting to the stimulus when the fixation input is set to zero, the inability to maintain this capacity resulted in the drop in accuracy on the overall model. Using the traditional cognitive ontology, unit 28 seems to control the models inhibition system or sustained attention, allowing the model to wait for a go signal to define when it should react and responding preemptively when this system is off. Response inhibition is a well-documented cognitive process that allows an individual to control their behavioural responses and impulses. In the brain, the process is associated with the prefrontal cortex, caudate nucleus and subthalamic nucleus [1], and damages to these regions result in a lack of control similar to the behaviour observed by the unitless model. Despite the complexity of the inhibitory system in animals not being comparable to the complexity of a 2 layered artificial neural network, the understanding of this process in the artificial system will express a mechanistic solution to inhibition control and might provide hypothesis for the biological system.
Unit 28 has a clear role in the models outcome but there are certainly other units that contribute to the following of the fixation input. Furthermore, there are other units, such as unit 93 and 160, that exhibit high variance across trials and report a function apparently associated with a process common for every trial of RT tasks. This is not the general case and most do not have transparent functional mappings, with a large number of units in the first layer reporting low to no variance. This does not mean that they have no function, as the model applies backpropagation while learning to maximise every units contribution to the final outcome. What is certainly the case in most of the tasks functions is that their processes are distributed across a number of different units, each contributing to the correct execution. This process of cooperation is common in trained ANNs and is also displayed in brain structure as the processes depend not on a single region but on networks of brain structures.

4.4. Biological comparisons and limitations

This project serves as a first step towards a higher goal of simulating the brain mechanisms in solving cognitive processes. Despite this, it is still hard to draw a clear parallel between the artificial model and the biological system.
There are a number of explanations for a lack of representations for the computational and neural description. The most intuitive one would be to disregard the artificial model as a bad proxy for the brain. Although this is a possibility, there are some other considerations that should be taken into account.
The dataset that serves as input and output for the model to learn the cognitive tasks might not represent correctly the given cognitive processes,

despite being influenced by the work of Yang et al. [14]. They tried to model the interaction between an animal and the environment when solving these tasks in a simplified manner so that the information could be provided to the artificial model. If so, a different task setup might result in a more significant correlation between models.

There might also be a lack of representation similarity due to the low number of cognitive tasks being solved by the artificial model. The animal brain is able to solve a number of cognitive tasks and so, the brain is optimized to efficiently solve each of them. By using a larger number of cognitive tasks, the model will be forced to optimize its parameters for different processes, resulting in compositionality of tasks and, possibly, in higher similarity to equivalent brain mappings.
Furthermore, although the model is supposed to represent a simplified version of the brain, there are significant structural dissimilarities that can be addressed. The layers in this model are densely connected (i.e. each unit connects to every unit of the following layer), which is not verified in the brain. The supervised learning mechanism and the backpropagation algorithm are not biologically plausible and the artificial model does not account for neural mechanisms such as neuromodulation or spiking. It is also important to note that the model does not intend to mimic directly neurons or networks, but only the mechanisms used by the brain despite of the level of analysis. Finally, although backpropagation does not appear biologically logical, the mechanisms the brain uses to learn are still not fully understood and it is impossible to tell if the backpropagation algorithm does not present the same results as the brain mechanism.

5. Conclusion

This projects main contribution is demonstrating an ANN architecture capable of effectively solving six cognitive tasks with an accuracy rate of 93% developed for this work. There is no literature available on using LSTM units to simulate multiple cognitive tasks as this project does, allowing it to deal with longer dependencies, retaining information for a larger number of iterations, useful features when solving cognitive tasks as most require the engagement of working memory for its successful execution in biological systems. On that note, LSTM units characteristically store a memory value that is recurrently passed along iterations called cell state. This state is seen as a proxy of memory in biological systems, so its analysis allows for interesting conclusions.
In a more general manner, this work operates as a first step towards mapping the artificial task representation to its biological counterpart and consequently, mapping the cognitive paradigms to the brain networks activations and their implied mechanisms. It is through understanding the mechanisms behind cognitive processes that an effective cognitive ontology can be built without disregarding any level of brain studies. If, with further work, the dissimilarities in representations of different tasks between the biological and artificial systems are still present, the artificial model will, nonetheless, convey a successful mechanistic solution for the cognitive processes, a concrete step towards demystifying intelligence.

5.1. Future Work

This work opens way for further analysis and new projects with the goal of better understanding the artificial model and possibly, the brain. There are several model optimizations that can be considered when the model analysis is taken into account. The RT tasks account for a disproportionate amount of the models variance and so it should be considered if its removal or substitution by different tasks in the training set would require a smaller number of units to be successfully trained. Furthermore, future work could investigate how units would behave mechanistically, by applying an analysis similar to the one reported for unit 28. Another approach could be to delineate a simplified version of the RT tasks which would prove easier for the model to learn.
As unit 28 was analysed attentively, there are a number of other units that can have a significant and transparent role for the models success. The same method of analysis can be undertaken for different units and, ultimately, even the units whose functions are more abstract or result in a small contribution towards a given process, should be analysed and understood in order to fully map out the models mechanistic solution for each of the cognitive tasks. A more targeted weight analysis with focus on the first seven input units (i.e. rule inputs and fixation) may allow further understanding on how each unit prioritizes its processes.
In order to avoid difficulties in drawing datasets that would correctly simulate cognitive tasks, a more reliable solution would be to collect fMRI data of subjects solving the exact same dataset that serves as input for the model. Psychlab [7] software facilitates this process as it creates an open-source virtual environment that can also be fed into a neural network, successfully replicating the same environment provided for both the user and the machine. Additionally, the model can be adapted towards a more biologically plausible learning method with reinforcement learning, where the model only knows if it has successfully executed the trial when

it is finished. Currently, with supervised learning, the models weights are updated every time-step depending on following the correct solution or not. Finally, a larger number of cognitive tasks should be considered and trained. The source code that generates the dataset was built to allow flexibility for the user to implement different tasks allowing for desirable reaction manipulation and two different modalities that are paramount for multisensory or decision-making tasks. The larger number of tasks learned will force the model to efficiently allocate units function, resulting in a mechanism that competently solves cognitive processes.

Acknowledgements
The author would like to thank Professor Robert Leech, Doctor Romy Lorenz and Professor Rita Nunes for all the insightful contributions to this work. The author would also like to express his gratitude to PhD student Sebastian Popescu, whose day-to-day contribution were invaluable for the success of this project.

References

[1] F. Carnevale, V. DeLafuente, R. Romo, O. Barak, and N. Parga. Dynamic Control of Response Criterion in Premotor Cortex during Perceptual Detection under Temporal Uncertainty. *Neuron*, 86(4):1067–1077, 2015.

[2] R. Caruana. Multitask Learning. *Learn. to Learn*, 75:41–75, 1997.

[3] I. Eisenberg, P. Bissett, A. Z. Enkavi, J. Li, D. MacKinnon, L. Marsch, and R. Poldrack. Uncovering mental structure through data-driven ontology discovery. *PsyArXiv*, page n. pag., 2018.

[4] J. M. Findlay. The Visual Stimulus for Saccadic Eye Movements in Human Observers. *Perception*, 9(1):7–21, feb 1980.

[5] S. Funahashi and C. J. Bruce. Mnemonic coding of visual space in the monkey ' s dorsolateral prefrontal cortex. *J. Neurophysiol.*, 61(2):331–349, 1989.

[6] D. P. Kingma and J. L. Ba. Adam : a method for stochastic optimization. *ICLR*, pages 1–15, 2015.

[7] J. Z. Leibo, C. d. M. D'Autume, D. Zoran, D. Amos, C. Beattie, K. Anderson, A. G. Castañeda, M. Sanchez, S. Green, A. Gruslys, S. Legg, D. Hassabis, and M. M. Botvinick. Psychlab: A Psychology Laboratory for Deep Reinforcement Learning Agents. *arXiv*, pages 1–28, 2018.

[8] T. Miconi. Biologically plausible learning in recurrent neural networks reproduces neural dynamics observed during cognitive tasks. *Elife*, 6:1–22, 2017.

[9] D. P. Munoz and S. Everling. Look away: the anti-saccade task and the voluntary control of eye movement. *Nat. Rev. Neurosci.*, 5:218, mar 2004.

[10] L. Pessoa. Understanding Brain Networks. *Phys. Life Rev.*, 11(3):400–435, 2015.

[11] L. J. P. Van Der Maaten and G. E. Hinton. Visualizing high-dimensional data using t-sne. *J. Mach. Learn. Res.*, 9:2579–2605, 2008.

[12] D. A. Weiskopf. Integrative Modeling and the Role of Neural Constraints. *Philos. Sci.*, 83(5):674–685, 2016.

[13] G. R. Yang, I. Ganichev, X.-J. Wang, J. Shlens, and D. Sussillo. A dataset and architecture for visual reasoning with a working memory. 2018.

[14] G. R. Yang, H. F. Song, W. T. Newsome, and X.-J. Wang. Clustering and compositionality of task representations in a neural network trained to perform many cognitive tasks. *bioRxiv*, page 183632, 2017.